



**National Institute of  
Standards and Technology**  
U.S. Department of Commerce

Special Publication 800-126  
Revision 1 (**Second Public Draft**)

---

# The Technical Specification for the Security Content Automation Protocol (SCAP): SCAP Version 1.1 (DRAFT)

---

## **Recommendations of the National Institute of Standards and Technology**

---

Stephen Quinn  
David Waltermire  
Christopher Johnson  
Karen Scarfone  
John Banghart

NIST Special Publication 800-126  
Revision 1 (Second Public Draft)

# The Technical Specification for the Security Content Automation Protocol (SCAP): SCAP Version 1.1 (Draft)

*Recommendations of the National  
Institute of Standards and Technology*

Stephen Quinn  
David Waltermire  
Christopher Johnson  
Karen Scarfone  
John Banghart

---

## C O M P U T E R   S E C U R I T Y

---

Computer Security Division  
Information Technology Laboratory  
National Institute of Standards and Technology  
Gaithersburg, MD 20899-8930

May 2010



**U.S. Department of Commerce**

Gary Locke, Secretary

**National Institute of Standards and Technology**

Dr. Patrick D. Gallagher, Director

## Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analysis to advance the development and productive use of information technology. ITL's responsibilities include the development of technical, physical, administrative, and management standards and guidelines for the cost-effective security and privacy of sensitive unclassified information in Federal computer systems. This Special Publication 800-series reports on ITL's research, guidance, and outreach efforts in computer security and its collaborative activities with industry, government, and academic organizations.

**National Institute of Standards and Technology Special Publication 800-126, Revision 1 (Draft)**  
**71 pages (Apr. 2010)**

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

## Acknowledgments

The authors, Stephen Quinn, David Waltermire, Christopher Johnson, Karen Scarfone, and John Banghart of the National Institute of Standards and Technology (NIST), wish to thank their colleagues who reviewed drafts of this document and contributed to its technical content. The authors would like to acknowledge Paul Bartock of the National Security Agency (NSA); Jeff Ito, Matt Kerr, Shane Shaffer, and Greg Witte of G2; Andy Bove of SecureAcuity; Jim Ronayne of Cobham; Adam Halbardier, Paul Cichonski, Angela Orebaugh, and Victoria Thompson of Booz Allen Hamilton; and Jon Baker, Drew Buttner, and Maria Casipe of the MITRE Corporation for their keen and insightful assistance throughout the development of the document.

## Trademark Information

OVAL and CVE are registered trademarks, and CCE, CPE, and OCIL are trademarks, of The MITRE Corporation.

XCCDF and SCAP are trademarks of the National Institute of Standards and Technology (NIST).

Windows XP, Windows Vista, and Windows Server 2003 are registered trademarks of Microsoft Corporation

All other registered trademarks or trademarks belong to their respective organizations.

## Table of Contents

<b>Executive Summary .....</b>	<b>ES-1</b>
<b>1. Introduction .....</b>	<b>1-1</b>
1.1 Authority .....	1-1
1.2 Purpose and Scope .....	1-1
1.3 Audience .....	1-1
1.4 Document Structure .....	1-2
1.5 Document Conventions .....	1-2
<b>2. Overview of SCAP 1.1 .....</b>	<b>2-1</b>
<b>3. Basics of SCAP Components.....</b>	<b>3-1</b>
3.1 Languages .....	3-1
3.1.1 Extensible Configuration Checklist Description Format (XCCDF) 1.1.4 .....	3-1
3.1.2 Open Vulnerability and Assessment Language (OVAL) 5.6 .....	3-3
3.1.3 Open Checklist Interactive Language (OCIL) 2.0 .....	3-5
3.2 Enumerations .....	3-6
3.2.1 Common Platform Enumeration (CPE) 2.2.....	3-6
3.2.2 Common Configuration Enumeration (CCE) 5.....	3-7
3.2.3 Common Vulnerabilities and Exposures (CVE) .....	3-8
3.3 Common Vulnerability Scoring System (CVSS) 2.0.....	3-8
<b>4. SCAP General Requirements and Conventions.....</b>	<b>4-1</b>
4.1 Support for Legacy SCAP Versions .....	4-1
4.2 XCCDF Conventions and Requirements .....	4-1
4.2.1 Metadata Elements .....	4-1
4.2.2 Use of CPE Names and CPE Language Statements .....	4-2
4.2.3 The <xccdf:Benchmark> Element .....	4-3
4.2.4 The <xccdf:Profile> Element .....	4-4
4.2.5 The <xccdf:Rule> Element .....	4-4
4.2.6 Allowed Check System Usage .....	4-6
4.2.7 XCCDF Test Results.....	4-11
4.3 OVAL Conventions and Requirements .....	4-13
4.3.1 Supported Previous Versions of OVAL (5.3, 5.4, and 5.5) .....	4-14
4.3.2 Support for Deprecated Constructs in OVAL .....	4-14
4.3.3 OVAL Schema Specification .....	4-14
4.3.4 OVAL Results .....	4-14
4.4 OCIL Conventions .....	4-16
4.5 CPE Conventions .....	4-16
4.6 CCE Conventions .....	4-17
4.7 CVE Conventions .....	4-17
4.8 CVSS Conventions.....	4-18
<b>5. SCAP Use Case Requirements.....</b>	<b>5-1</b>
5.1 SCAP Data Streams.....	5-1
5.1.1 SCAP Source Data Streams .....	5-1
5.1.2 SCAP Result Data Streams .....	5-1
5.2 SCAP Configuration Verification .....	5-2
5.2.1 SCAP Configuration Verification Source Data Streams .....	5-2

5.2.2	SCAP Configuration Verification Results .....	5-3
5.3	SCAP Vulnerability Assessment .....	5-4
5.3.1	SCAP Vulnerability Assessment Using XCCDF Source Data Streams .....	5-4
5.3.2	SCAP Vulnerability Assessment Results Using XCCDF .....	5-5
5.3.3	SCAP Vulnerability Assessment Using Standalone OVAL Source(s) .....	5-5
5.3.4	OVAL Definitions and Vulnerability Assessment .....	5-5
5.3.5	SCAP Vulnerability Assessment Results Using Standalone OVAL .....	5-6
5.4	Patch Validation .....	5-6
5.4.1	SCAP Patch Validation Using Standalone OVAL .....	5-6
5.4.2	SCAP Patch Validation Results Using Standalone OVAL .....	5-6
5.5	SCAP Inventory Collection .....	5-6
5.5.1	SCAP Inventory Collection Data Sources .....	5-7
5.5.2	SCAP Inventory Collection Data Results .....	5-7

## List of Appendices

<b>Appendix A— Acronyms and Abbreviations .....</b>	<b>A-1</b>
<b>Appendix B— References and Other Resources .....</b>	<b>B-1</b>
<b>Appendix C— SCAP Compliance Verification Data Stream Example .....</b>	<b>C-1</b>
C.1 XCCDF Benchmark .....	C-1
C.2 OVAL Compliance .....	C-5
C.3 OVAL Patch .....	C-10
C.4 OCIL Questionnaire .....	C-18
C.5 CPE Dictionary .....	C-18
C.6 CPE Inventory .....	C-19

## List of Figures

Figure A-1. Example CCE Entry .....	3-7
Figure C-1 example-winxp-xccdf.xml .....	C-1
Figure C-2 example-winxp-oval.xml .....	C-5
Figure C-3 example-winxp-patches.xml .....	C-10
Figure C-4. example-winxp-ocil.xml .....	C-18
Figure C-5. example-winxp-cpe-dictionary.xml .....	C-18
Figure C-6. example-winxp-cpe-oval.xml .....	C-19

## List of Tables

Table 1-1. Conventional XML Mappings .....	1-3
Table 3-1. XCCDF Rule Sample Data .....	3-2
Table 3-2. XCCDF Value Statement Sample Data .....	3-3

Table 3-3. OVAL Definition Sample Data .....	3-4
Table 3-4. OCIL Questionnaire Sample Data .....	3-6
Table 4-1. Use of Dublin Core Terms in XCCDF Metadata .....	4-4
Table 4-2. XCCDF-OVAL Data Export Matching Constraints .....	4-8
Table 4-3. Deriving XCCDF Rule Results from OVAL Definition Results.....	4-9
Table 4-4. XCCDF-OCIL Data Export Matching Constraints .....	4-11
Table 4-5. Deriving XCCDF Rule Results from OCIL Questionnaire Results.....	4-11
Table 4-6. XCCDF Fact Descriptions .....	4-12
Table 5-1. SCAP Configuration Verification Data Sources .....	5-2
Table 5-2. SCAP Configuration Verification Data Results .....	5-3
Table 5-3. SCAP Vulnerability Assessment Data Sources .....	5-4
Table 5-4. SCAP Vulnerability Assessment Results .....	5-5
Table 5-5. SCAP Inventory Collection .....	5-7
Table 5-6. SCAP Inventory Collection Results .....	5-7

## Executive Summary

The Security Content Automation Protocol (SCAP) is a suite of specifications that standardize the format and nomenclature by which security software products communicate software flaw and security configuration information<sup>1</sup>. SCAP is a multi-purpose protocol that supports automated vulnerability and patch checking, technical control compliance activities, and security measurement. Goals for the development of SCAP include standardizing system security management, promoting interoperability of security products, and fostering the use of standard expressions of security content.

This document defines the technical composition of SCAP Version 1.1 as comprised of seven specifications—eXtensible Configuration Checklist Description Format (XCCDF), Open Vulnerability and Assessment Language (OVAL®), Open Checklist Interactive Language (OCIL), Common Platform Enumeration (CPE™), Common Configuration Enumeration (CCE™), Common Vulnerabilities and Exposures (CVE®), and Common Vulnerability Scoring System (CVSS)—and their interrelationships. These specifications are grouped into the following three categories:

- **Languages.** The SCAP languages provide standard vocabularies and conventions for expressing security policy, technical check mechanisms, and assessment results.
- **Enumerations.** Each SCAP enumeration defines a standard nomenclature (naming format) and an official dictionary or list of items expressed using that nomenclature. For example, CVE provides a dictionary of publicly known information security vulnerabilities and exposures.<sup>2</sup>
- **Measurement and scoring systems.** In SCAP, this refers to evaluating specific characteristics of a vulnerability and, based on those characteristics, generating a score that reflects the vulnerability's severity.

SCAP utilizes software flaw and security configuration standard reference data, also known as *SCAP content*. This reference data is provided by the National Vulnerability Database (NVD),<sup>3</sup> which is managed by NIST and sponsored by the Department of Homeland Security (DHS).

This publication defines SCAP Version 1.1 in terms of both its component specifications and the requirements for SCAP content, and also describes the details of how the elements of SCAP interoperate. The technical specification describes the requirements and conventions that are to be employed to ensure the consistent and accurate exchange of SCAP content and the ability to reliably use the content with SCAP validated products.

The U.S. Federal Government, in cooperation with academia and private industry, is adopting SCAP and encourages its use in support of security automation activities and initiatives.<sup>4</sup> SCAP is achieving widespread adoption by major software and hardware manufacturers and has become a significant component of large information security management and governance programs. The protocol is expected to evolve and expand in support of the growing needs to define and measure effective security controls, assess and monitor ongoing aspects of that information security, and successfully manage systems in accordance with risk management frameworks such as NIST Special Publication 800-53<sup>5</sup>, Department of

<sup>1</sup> Products implementing SCAP can also be used to support non-security use cases such as configuration management and software inventory.

<sup>2</sup> <http://cve.mitre.org/>

<sup>3</sup> The National Vulnerability Database can be found at <http://nvd.nist.gov/>.

<sup>4</sup> Refer to <http://www.whitehouse.gov/omb/memoranda/fy2008/m08-22.pdf>.

<sup>5</sup> The Risk Management Framework is described in Section 3.0 of NIST Special Publication 800-53, available at <http://csrc.nist.gov/publications/nistpubs/800-53-Rev3/sp800-53-rev3-final-errata.pdf>.



Defense (DoD) Instruction 8500.2, and the Payment Card Industry (PCI) framework.

By detailing the specific and appropriate usage of the SCAP 1.1 components and their interoperability, NIST encourages the creation of reliable and pervasive SCAP content and the development of a wide array of products that leverage SCAP capabilities. The use cases described in this document do not represent an exhaustive list of all possible applications of SCAP.

Organizations that use SCAP 1.1 or develop SCAP 1.1-based content or products should implement the following recommendations:

**Follow the requirements listed in this document and in the associated component specifications.**

Organizations should ensure that their use of SCAP 1.1 is compliant with the requirements detailed in each component specification and the information presented in this document. If requirements are in conflict between component specifications, this document will provide clarification. If a component specification is in conflict with this document, the requirements in this document take precedence.

**When creating SCAP content, adhere to the conventions specified in this document.**

Security products and checklist authors assemble content from SCAP data repositories to create viable SCAP-expressed security guidance. A security configuration checklist that documents desired security configuration settings, installed patches, and other system security elements using SCAP in a standardized format is known as an SCAP-expressed checklist. Such a checklist would use XCCDF to describe the checklist, CCE to identify security configuration settings to be addressed or assessed, and CPE to identify platforms for which the checklist is valid. The use of CCE and CPE entries within XCCDF checklists is an example of an SCAP convention—a requirement for valid SCAP usage. These conventions are considered part of the definition of SCAP 1.1. Organizations producing SCAP content should adhere to these conventions to ensure the highest degree of interoperability.

## **1. Introduction**

### **1.1 Authority**

The National Institute of Standards and Technology (NIST) developed this document in furtherance of its statutory responsibilities under the Federal Information Security Management Act (FISMA) of 2002, Public Law 107-347.

NIST is responsible for developing standards and guidelines, including minimum requirements, for providing adequate information security for all agency operations and assets; but such standards and guidelines shall not apply to national security systems. This guideline is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130, Section 8b(3), “Securing Agency Information Systems,” as analyzed in A-130, Appendix IV: Analysis of Key Sections. Supplemental information is provided in A-130, Appendix III.

This guideline has been prepared for use by Federal agencies. It may be used by nongovernmental organizations on a voluntary basis and is not subject to copyright, though attribution is desired.

Nothing in this document should be taken to contradict standards and guidelines made mandatory and binding on Federal agencies by the Secretary of Commerce under statutory authority, nor should these guidelines be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the OMB, or any other Federal official.

### **1.2 Purpose and Scope**

This document provides the definitive technical specification for Version 1.1 of the Security Content Automation Protocol (SCAP). SCAP (pronounced S-CAP) consists of a suite of specifications for standardizing the format and nomenclature by which security software communicates information about software flaws and security configurations. This document describes the basics of the SCAP component specifications and their interrelationships, the characteristics of SCAP content, as well as SCAP requirements not defined in the individual component specifications.

The scope of this document is limited to SCAP Version 1.1. Other versions of SCAP and the component specifications, including emerging specifications and future versions of SCAP, are not addressed here. Future versions of SCAP will be defined in distinct revisions of this document, each clearly labeled with a document revision number and the appropriate SCAP version number.

### **1.3 Audience**

This document is intended for three primary audiences:

- Content authors and editors seeking guidance to ensure that the SCAP content they produce operates correctly, consistently, and reliably in SCAP products.
- Software developers and system integrators seeking to create, use, or exchange SCAP content in their products or service offerings.
- Product developers preparing for SCAP validation at an accredited independent testing laboratory.

## 1.4 Document Structure

The remainder of this document is organized into the following four major sections:

- Section 2 defines SCAP 1.1 and explains the purpose of SCAP.
- Section 3 presents basic information on the specifications comprising SCAP 1.1.
- Section 4 defines conventions and requirements for using SCAP to achieve interoperability of content and products.
- Section 5 presents the concept of an SCAP data stream and describes use cases that demonstrate effective and compliant implementations of SCAP.

The document also contains appendices with supporting material:

- Appendix A contains an acronym and abbreviation list.
- Appendix B lists references and other resources related to SCAP 1.1 and its component specifications.
- Appendix C provides an example of an SCAP data stream.

## 1.5 Document Conventions

Some of the requirements and conventions used in this document reference XML content. These references come in two forms, inline and indented. An example of an inline reference is

“A `<cpe_dict:cpe-item>` may contain `<cpe_dict:check>` elements that reference OVAL Definitions”.

In this example the notation `<cpe_dict:cpe-item>` can be replaced by the more verbose equivalent “the XML element whose qualified name is `cpe_dict:cpe-item`”. An even more verbose equivalent is “the XML element in the namespace ‘`http://cpe.mitre.org/dictionary/2.0`’ whose local name is `cpe-item`”.

An example of an indented reference is:

```
“References to OVAL Definitions are expressed using the following format:  
<cpe_dict:check system=  
  "http://oval.mitre.org/XMLSchema/oval-definitions-5"  
  href="Oval_URL">[Oval_inventory_definition_id]</cpe_dict:check>”.
```

The general convention used when describing XML attributes within this document is to reference the attribute as well as its associated element including the namespace alias, employing the general form:

“@*attributeName* for the `<prefix:localName>`”.

Indented references are intended to represent the form of actual XML content. Indented references represent literal content by the use of a *fixed-length font*, and parametric (freely replaceable) content by the use of an *italic font*. Square brackets ‘[ ]’ are used to designate optional content. Thus “[*Oval\_inventory\_definition\_id*]” designates optional parametric content.

Both inline and indented forms use qualified names to refer to specific XML elements. A qualified name

associates a named element with a namespace. The namespace identifies the specific XML schema that defines (and consequently may be used to validate) the syntax of the element instance. A qualified name declares this schema to element association using the format ‘*prefix:element-name*’. The association of prefix to namespace is defined in the metadata of an XML document and generally will vary from document to document. In this specification, the conventional mappings listed in Table 1-1 are used.

**Table 1-1. Conventional XML Mappings**

Prefix	Namespace URI	Schema
cpe_dict	<a href="http://cpe.mitre.org/dictionary/2.0">http://cpe.mitre.org/dictionary/2.0</a>	CPE Dictionaries
cpe	<a href="http://cpe.mitre.org/language/2.0">http://cpe.mitre.org/language/2.0</a>	Embedded CPE references
nvd	<a href="http://scap.nist.gov/schema/feed/vulnerability/2.0">http://scap.nist.gov/schema/feed/vulnerability/2.0</a>	Base schema for NVD data feeds
cve	<a href="http://scap.nist.gov/schema/vulnerability/0.4">http://scap.nist.gov/schema/vulnerability/0.4</a>	NVD/CVE data feed elements and attributes
cvss	<a href="http://scap.nist.gov/schema/cvss-v2/0.2">http://scap.nist.gov/schema/cvss-v2/0.2</a>	NVD/CVSS data feed elements and attributes
dc	<a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a>	Simple Dublin Core elements
xccdf	<a href="http://checklists.nist.gov/xccdf/1.1">http://checklists.nist.gov/xccdf/1.1</a>	XCCDF policy documents
xml	<a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a>	Common XML attributes
inter	<a href="http://www.mitre.org/ocil/2">http://www.mitre.org/ocil/2</a>	OCIL elements and attributes
oval	<a href="http://oval.mitre.org/XMLSchema/oval-common-5">http://oval.mitre.org/XMLSchema/oval-common-5</a>	Common OVAL elements and attributes
oval-def	<a href="http://oval.mitre.org/XMLSchema/oval-definitions-5">http://oval.mitre.org/XMLSchema/oval-definitions-5</a>	OVAL Definitions
xxxx-def	<a href="http://oval.mitre.org/XMLSchema/oval-definitions-5#xxxx">http://oval.mitre.org/XMLSchema/oval-definitions-5#xxxx</a>	OVAL elements and attributes specific to an OS, Hardware, or Application type xxxx <sup>6</sup>
oval-res	<a href="http://oval.mitre.org/XMLSchema/oval-results-5">http://oval.mitre.org/XMLSchema/oval-results-5</a>	OVAL results
oval-sc	<a href="http://oval.mitre.org/XMLSchema/oval-system-characteristics-5">http://oval.mitre.org/XMLSchema/oval-system-characteristics-5</a>	OVAL system characteristics
xxxx-sc	<a href="http://oval.mitre.org/XMLSchema/oval-system-characteristics-5#xxxx">http://oval.mitre.org/XMLSchema/oval-system-characteristics-5#xxxx</a>	OVAL system characteristic elements and attributes specific to an OS, Hardware, or Application type xxxx
oval-var	<a href="http://oval.mitre.org/XMLSchema/oval-variables-5">http://oval.mitre.org/XMLSchema/oval-variables-5</a>	The elements, types, and attributes that compose the core schema for encoding OVAL Variables. This schema is provided to give structure to any external variables and their values that an OVAL Definition is expecting.
sch	<a href="http://purl.oclc.org/dsdl/schematron">http://purl.oclc.org/dsdl/schematron</a>	Schematron validation scripts
ds	<a href="http://www.w3.org/2000/09/xmldsig#">http://www.w3.org/2000/09/xmldsig#</a>	Interoperable XML digital signatures

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in Request for Comment (RFC) 2119.<sup>7</sup>

<sup>6</sup> The types supported by OVAL 5.3 include the AIX, CATOS, ESX, FREE BSD, HP-UX, IOS, LINUX, PIXOS, SOLARIS, UNIX, WINDOWS, INDEPENDENT (common) operating systems, and APACHE application.

<sup>7</sup> RFC 2119, “Key words for use in RFCs to Indicate Requirement Levels”, is available at <http://www.ietf.org/rfc/rfc2119.txt>.

## 2. Overview of SCAP 1.1

NIST Special Publication (SP) 800-117, *Guide to Adopting and Using the Security Content Automation Protocol*,<sup>8</sup> defines SCAP as being comprised of two major elements. [BAR09] First, SCAP is a protocol—a suite of specifications that standardize the format and nomenclature by which security software communicates information about publicly known software flaws and security configurations annotated with common identifiers and embedded in XML. Second, SCAP also utilizes software flaw and security configuration standard reference data, also known as *SCAP content*. This reference data is provided by the National Vulnerability Database (NVD),<sup>9</sup> which is managed by NIST and sponsored by the Department of Homeland Security (DHS). SCAP can be used for several purposes, including automating vulnerability checking, technical control compliance activities, and security measurement. The U.S. Federal Government, in cooperation with academia and private industry, is adopting SCAP and is encouraging widespread support of it.

This document defines Version 1.1 of SCAP in terms of both its component specifications and the requirements for SCAP content. As stated in the Executive Summary, organizations that use SCAP 1.1 should ensure that their use of it is compliant with the requirements detailed in each component specification and the information presented in this document. If requirements are in conflict between component specifications, this document will provide clarification. If a component specification is in conflict with this document, the requirements in this document SHALL take precedence.

SCAP 1.1 uses the following specifications:

- Extensible Configuration Checklist Description Format (XCCDF) 1.1.4, a language for authoring security checklists/benchmarks and for reporting results of checklist evaluation [QUI08]
- Open Vulnerability and Assessment Language (OVAL) 5.6, a language for representing system configuration information, assessing machine state, and reporting assessment results
- Open Checklist Interactive Language (OCIL) 2.0, a language for representing security checks that requires human feedback
- Common Platform Enumeration (CPE) 2.2, a nomenclature and dictionary of hardware, operating systems, and applications [BUT09]
- Common Configuration Enumeration (CCE) 5, a nomenclature and dictionary of security software configurations
- Common Vulnerabilities and Exposures (CVE), a nomenclature and dictionary of security-related software flaws<sup>10</sup>
- Common Vulnerability Scoring System (CVSS) 2.0, an open specification for measuring the relative severity of software flaw vulnerabilities [MEL07].

Section 3 presents detailed information on each of these specifications and provides examples of how these components are used in context.

Security products and checklist authors assemble content from SCAP data repositories to create viable SCAP-expressed security guidance. As stated in the Executive Summary, a security configuration

<sup>8</sup> NIST SP 800-117 is available at <http://csrc.nist.gov/publications/PubsDrafts.html#SP-800-117>

<sup>9</sup> <http://nvd.nist.gov/>

<sup>10</sup> CVE does not have a version number.

checklist that documents desired security configuration settings, installed patches, and other system security elements using SCAP in a standardized format is known as an SCAP-expressed checklist. Such a checklist would use XCCDF to describe the checklist, CCE to identify security configuration settings to be addressed or assessed, and CPE to identify platforms for which the checklist is valid. The use of CCE and CPE entries within XCCDF checklists is an example of an SCAP convention—a requirement for valid SCAP usage. These conventions are considered part of the definition of SCAP 1.1 and are described in Sections 3, 4, and 5 of this document. Organizations producing SCAP content should adhere to these conventions to ensure the highest degree of interoperability.

SCAP revisions are managed through a coordinated process defined within the SCAP Release Cycle.<sup>11</sup> The release cycle workflow manages changes related to SCAP specifications and validation processes including the addition of new specifications or updates to existing specifications. This process encourages community involvement, promotes transparency and awareness regarding proposed changes, and affords ample lead-time to prepare for pending changes.

---

<sup>11</sup> SCAP Release Cycle, <http://scap.nist.gov/timeline.html>

### 3. Basics of SCAP Components

SCAP 1.1 is comprised of the specifications referenced in Section 2: XCCDF, OVAL, OCIL, CPE, CCE, CVE, and CVSS. These specifications are grouped into the following three categories:

1. **Languages.** SCAP languages provide a standardized means for identifying what is to be evaluated and for expressing how to check system state.
2. **Enumerations.** SCAP enumerations provide a standardized nomenclature (naming format) and an associated dictionary of items expressed using that nomenclature. For example, CVE provides a dictionary of publicly known information security vulnerabilities and exposures.<sup>12</sup>
3. **Measurement and scoring systems.** SCAP vulnerability measurement and scoring systems provide the ability within SCAP to measure and evaluate specific vulnerability characteristics to derive a vulnerability severity score.

This section provides an introduction to the SCAP component specifications in each of these categories.

#### 3.1 Languages

This section describes the three language specifications in SCAP 1.1: XCCDF 1.1.4, OVAL 5.6, and OCIL 2.0. The following sections describe the purpose and primary logical concepts for each specification and provide examples. Products may be required to process previous revisions of these specifications. See Section 4 for additional details on the specifications (Section 4.2 for XCCDF, Section 4.3 for OVAL, and Section 0 for OCIL).

##### 3.1.1 Extensible Configuration Checklist Description Format (XCCDF) 1.1.4

XCCDF 1.1.4 is a specification language for expressing security configuration checklists, vulnerability alerts, and other related documents. The specification is designed to support information interchange, document generation, organizational and situational tailoring, automated compliance testing, and compliance scoring. An XCCDF document represents a structured collection of system assessment rules for some set of target systems. The specification also defines a data model and format for storing results of assessing an XCCDF benchmark. The intent of XCCDF is to provide a uniform means of expressing security checklists and the results of checklist evaluation.

An XCCDF document is composed of one or more XCCDF rules. An XCCDF rule is a high-level definition of a technical check on a system. A rule does not directly specify how a check should be performed, but instead points to other XML documents (such as OVAL or OCIL Definition files) that contain the actual instructions for performing the check. Table 3-1 shows sample values from an XCCDF rule. This particular rule is for ensuring that the minimum password length is set to at least eight characters. The System Check section of the rule specifies the OVAL Definition example presented in Table 3-3.

---

<sup>12</sup> <http://cve.mitre.org/>

**Table 3-1. XCCDF Rule Sample Data**

Rule Field	Explanation	Sample Data
Rule ID	The identifier for this rule	MinimumPasswordLength-8
Title	The title for the rule	Minimum Password Length = 8
Description	The description of the rule	This setting specifies the minimum length of a password in characters. The rationale behind this setting is that longer passwords are more difficult to guess and crack than shorter passwords. The downside is that longer passwords are often more difficult for users to remember.
References	References to checklists and other documents that contain requirements to which this rule maps—in this case, the IA-5 (Authenticator Management) control from NIST SP 800-53	IA-5 ( <a href="http://csrc.nist.gov/publications/nistpubs/800-53-Rev3/sp800-53-rev3-final-errata.pdf">http://csrc.nist.gov/publications/nistpubs/800-53-Rev3/sp800-53-rev3-final-errata.pdf</a> )
Requires	The id of another Group or Rule in the Benchmark that should be selected for this Rule to be applied and scored properly. In this case, the IA-5 group	IA-5
System	The XML check system to use during rule evaluation (usually the OVAL Language)	<a href="http://oval.mitre.org/XMLSchema/oval-definitions-5">http://oval.mitre.org/XMLSchema/oval-definitions-5</a>
OVAL Definition File Reference	Name of the OVAL Definition file	WindowsXP-SP800-68.xml
OVAL Definition ID	The identifier of the OVAL Definition to be used	oval:gov.nist.1:def:20

The number of rules appearing in a typical XCCDF document will vary depending upon the intended use case. The rules appearing in an XCCDF document may also be organized into multiple *XCCDF profiles* that specify collections of rules to be evaluated on particular types of systems. Profiles can be used to express multiple policies within a single benchmark document; allowing the benchmark author to publish technical security control settings tailored to the type of system or the environment in which the system is deployed. By creating a policy that corresponds to a particular set of requirements, such as those of the FISMA, the Defense Information Systems Agency's (DISA) Security Technical Implementation Guides (STIG), or the Health Insurance Portability and Accountability Act (HIPAA), the policy can be used to map those high-level requirements to the corresponding OVAL Definitions.

An XCCDF document can be further organized into one or more *XCCDF groups*. A group can contain one or more related rules or groups. Groups allow multiple rules to be enabled or disabled collectively instead of individually.

Another option involving XCCDF rules is to have user-definable values for certain rules, known as *XCCDF values*. Table 3-2 shows sample data from an XCCDF value statement. This particular value statement defines the duration of the account lockout (in minutes) that occurs after consecutive failed login attempts have exceeded a specific threshold. In this case, the value has been set to 15 minutes and the operator field specifies that the system setting for lockout duration is greater than or equal to this value. A checklist user may choose to alter or override this value in the profile(s) that reference this value to account for specific organizational policies.



**Table 3-2. XCCDF Value Statement Sample Data**

Rule Field	Explanation	Sample Data
Value ID	The identifier for this value	AccountLockoutDurationTime
Type	The type of the value (e.g., string, number)	Number
Operator	The comparison operator (in this case, the system's value for account lockout duration time must be greater than or equal to the specified value)	greater than or equal
Title	The title for the value	Account Lockout Duration Time
Description	The description of the value	This value specifies how long the user account should be locked out. This is often set to a low but substantial value (e.g., 15 minutes).
Question	Explanatory text that can be presented to the user when is customizing the checklist	Account lockout duration time (in minutes)
Value	The value assigned to the AccountLockoutDurationTime value	15
Default	A suggested default value number for checklist users' reference; not actually used when performing checks or applying configuration settings	15

### 3.1.2 Open Vulnerability and Assessment Language (OVAL) 5.6

OVAL is used to express standardized, machine-readable rules that can be used to assess the state of a system. Under SCAP, OVAL is commonly used to determine the presence of vulnerabilities and insecure configurations. A set of instructions used to check for a security problem, such as an incorrect minimum password length setting, is known as an *OVAL Definition*. A file containing one or more OVAL Definitions (often hundreds or even thousands) is known as an *OVAL Definition file*.

There are four classes of OVAL Definitions used in SCAP:<sup>13</sup>

- Vulnerability definitions, which define “the conditions that must exist on a computer for a specific vulnerability to be present”
- Patch definitions, which define “the conditions on a computer that determine whether a particular patch is appropriate for a system”
- Inventory definitions, which define “the conditions on a computer that determine whether a specific piece of software is installed on the system”
- Compliance definitions, which define “the conditions on a computer that determine compliance with a specific policy or configuration statement”.

Table 3-3 shows sample values that have been extracted from an actual OVAL compliance definition. Explanations of each value have also been provided. The definition ID, version, and class are standard fields that are part of every OVAL Definition. The exact types of information contained in the metadata

<sup>13</sup> These definitions are taken from the OVAL Web site's “Structure of the Language” page, located at <http://oval.mitre.org/language/about/structure.html>.

vary among definitions, but at a high level they explain the intent of the definition. The criteria provide the technical details of how the system will be checked for the items of interest, such as the presence of vulnerability or the value of a configuration setting. Each OVAL Definition has a single top-level criterion that can contain one or more sub-criteria. The operator associated with each criterion specifies how the results produced by the sub-criteria are combined (e.g., AND, OR).<sup>14</sup>

The example in Table 3-3 has two criteria. One of the criteria is an *OVAL Test*, which is a specific system check—in this case, that the system is configured to require a minimum password length of at least eight characters. The other criterion is actually another definition—in this case, an inventory definition that confirms that the target system is running Windows XP SP2 on a 32-bit architecture.

**Table 3-3. OVAL Definition Sample Data**

Definition Field	Explanation	Sample Data
ID	Identifier for this definition; must be globally unique	oval:gov.nist.1:def:20
Version	Version of the definition	1
Class	Defines the type of definition (e.g., compliance, inventory, patch, vulnerability)	Compliance
Metadata		
Title	Short description for the definition	Minimum Password Length of 8 Characters
Affected product	The operating system or application version(s) to which this definition is applicable	Microsoft Windows XP, SP2, 32 bit
References	References to checklists and other documents that contain requirements to which this definition maps	NIST SP800-68 Appendix A, 1.4b, <a href="http://csrc.nist.gov/itsec/download_WinXP.html">http://csrc.nist.gov/itsec/download_WinXP.html</a> DISA FSO Checklist, 5.4.1.3 DISA VMS 6XID V0001106 DISA PDI ID 1740
Description	Description for the definition	The minimum allowable password length is 8 characters
Criteria		
Definition reference	The identifier of another OVAL Definition, OVAL Definition references another OVAL Definition (extended definition)	oval:gov.nist.1:def:9
Definition comment	A brief explanation of what the definition addresses; in this case, it is used to determine if the target system is running Windows XP SP2 on a 32-bit architecture	Precondition 9: Windows family, Windows XP, SP2, 32 bit
Test reference	An identifier for an OVAL Test that is run when evaluating the OVAL Definition.	oval:gov.nist.1:tst:16
Test comment	A brief explanation of what the test addresses; in this case, it is used to determine if the target system requires a minimum password length of 8 characters	Minimum password length is 8 characters

<sup>14</sup> In the context of this SP800-126 publication, the words ‘criterion’ and ‘criteria’ are used properly; however, the reader should note that the actual OVAL element names are expressed using ‘criteria’

As the example in Table 3-3 shows, definitions often reference one or more tests. The instructions that comprise each test are also included in the OVAL Definition file. A test does not directly contain the technical details of checking the system but instead references other OVAL constructs. Typically, a test references an *OVAL Object*, which is a logical construct for a portion of the target system (e.g., password policy, file, Windows registry key), and an *OVAL State*, which is a particular check of the specified OVAL Object (e.g., verifying that the password policy requires a minimum password length of at least eight characters, verifying the existence of a file). An OVAL State can also reference one or more OVAL Variables, which are user-definable values (e.g., minimum password length value of eight). This modular approach introduces additional complexity but fosters reuse and allows OVAL Definitions to be used without requiring the details of test construction to be exposed. Individuals seeking detailed information can refer to the OVAL Definition file for the Definition, Test, Object, and State ID numbers, and instructions associated with each entity. More technical details on OVAL Definition files, including examples of the XML code for OVAL Definitions, are presented in Section 4. An OVAL Definition tutorial is also available from the OVAL Web site at <http://oval.mitre.org/language/about/definition.html>.

### 3.1.3 Open Checklist Interactive Language (OCIL) 2.0

OCIL is a framework for expressing security checks that cannot be evaluated without some human interaction or feedback. It is used to determine the state of a system by presenting one or more questionnaires to its intended users. The language includes constructs for questions, instructions for guiding users towards an answer, responses to questions, artifacts, and evaluation results.

An OCIL document may contain one or more questionnaires. A questionnaire represents a single, discrete check similar to an OVAL Definition. It contains references to test actions (or other questionnaires) that describe a sequence of questions that must be posed to users. Based on user responses to questions, the result of a test action is computed. Test action results are aggregated with a logical operator, AND/OR, producing the questionnaire's result.

Four types of questions are supported in OCIL: Boolean, choice, numeric, and string. Each question type is designed to handle a particular data expected to be a valid answer. A question may also be associated with a set of instructions that provides a step-by-step procedure on how a user might answer the question. When a user responds to a question, the user may be required to provide an artifact (e.g., file, text) to support the answer.

Variables may also be used to set values for choices in choice questions, specify pattern matches for string questions, or set numerical range matches for numerical questions.

Table 3-4 is an example of an OCIL questionnaire that checks whether IPv6 Protocol 41 has been blocked. A unique identifier is required for each questionnaire within the document. Metadata fields are also provided to describe the questionnaire, references, and processing instructions.

**Table 3-4. OCIL Questionnaire Sample Data**

Questionnaire Field	Explanation	Sample Data
ID	Questionnaire identifier as described within the OCIL file	ocil:mitre.org:questionnaire:1
Actions	A container for all test actions that must be evaluated for the questionnaire	Test Action Reference: ocil:mitre.org:testaction:61 (describes a sequence of one or more questions)
Metadata		
Title	Short heading or caption for the questionnaire	IPv6 Blocks Protocol 41
Description	Descriptive text that describes the questionnaire in more detail	This questionnaire checks if IPv6 Protocol 41 is blocked
References	References to checklists and other documents that contain requirements to which this questionnaire maps	CCE-1795; CCE-2865-4
Scope	Processing instruction that specifies whether evaluation should stop when a result can be computed or continue to ask the remaining questions	FULL (indicates that all questions must be answered before computing the questionnaire's result)
Child Only	Specifies whether the questionnaire should be treated as top-level or a child-level questionnaire	False (indicates that this is a top-level questionnaire)
Priority	Priority level of questionnaire relative to other questionnaires	HIGH (indicates that this questionnaire is of "high" importance relative to other questionnaires within the document)

## 3.2 Enumerations

This section describes the three enumeration specifications in SCAP 1.1: CPE 2.2, CCE 5, and CVE. SCAP enumerations typically consist of an identifier, an associated description or definition, and a list of supporting references. For each specification, the section describes its purpose and provides examples of entries. The section also explains the interdependencies between these specifications and other SCAP component specifications.

### 3.2.1 Common Platform Enumeration (CPE) 2.2

CPE 2.2 is a standard naming convention for operating systems, hardware, and applications. The purpose of CPE is to provide consistent names that can be shared by multiple parties and solutions to refer to the same specific platform type.<sup>15</sup>

<sup>15</sup> The MITRE Corporation maintains the CPE specification and NIST maintains the Official CPE Dictionary. More information on CPE is available at <http://cpe.mitre.org/>. The Official CPE Dictionary is available at <http://nvd.nist.gov/cpe.cfm>

The syntax of an individual CPE Name, as defined in Section 5 of the CPE 2.2 Specification, is as follows:

```
cpe:/{part}:{vendor}:{product}:{version}:{update}:{edition}:{language}
```

For example, "cpe:/o:redhat:enterprise\_linux:2.1::es " refers to Red Hat Enterprise Server 2.1. The "o" indicates that this CPE describes an operating system. In this example, the edition field is blank, indicating that this CPE refers to all editions of Red Hat Enterprise Server 2.1.

CPE Names are used in conjunction with many of the SCAP specifications to provide an association to asset-related information. CPE is used by SCAP in the following ways:

- **XCCDF** – In an XCCDF checklist, CPE Names can be used to identify the hardware or software platform to which an XCCDF object (e.g., benchmark, profile, group, rule) applies.
- **CCE** – CPE Names can be associated with configuration vulnerabilities to identify platforms covered by CCE technical mechanisms.
- **CVE** – CVEs are related to one or more product platforms expressed as CPEs. The mapping of CPEs to CVEs is performed by NVD analysts and is published in the NVD vulnerability data feed.

### 3.2.2 Common Configuration Enumeration (CCE) 5

The CCE 5 naming scheme is a dictionary of names for security configuration settings for deployed software. Each type of security-related configuration issue is assigned a unique identifier to facilitate fast and accurate correlation of configuration data across multiple information sources and products. The MITRE Corporation publishes the CCE list.<sup>16</sup>

There are five attributes in a CCE entry: a unique identifier number, a description of the configuration issue, logical parameters of the CCE, the associated technical mechanisms related to the CCE, and references to additional sources of information. Figure 3-1 provides an example of these attributes for a CCE 5 entry for Windows XP.

**Figure A-1. Example CCE Entry**

<b>CCE ID:</b>	<b>CCE-3108-8</b>
<b>Definition:</b>	<b>The correct service permissions for the Telnet service should be assigned.</b>
<b>Parameters:</b>	<b>(1) set of accounts (2) list of permissions</b>
Technical	
Mechanisms:	(1) set via Security Templates (2) defined by Group Policy
References:	Listed at <a href="http://cce.mitre.org/lists/cce_list.html">http://cce.mitre.org/lists/cce_list.html</a>

References to CCEs are used by some of the other SCAP specifications to provide an association to particular security configuration settings. In an XCCDF checklist, CCEs can be used to specify which security configuration settings are of interest (i.e., which settings should be checked). Similarly, OVAL uses CCE entries for the same purpose.

<sup>16</sup> See <http://cce.mitre.org/> for additional information.

### 3.2.3 Common Vulnerabilities and Exposures (CVE)

CVE is a dictionary of unique, common names for publicly known software flaws.<sup>17</sup> This common naming convention allows sharing of data within and among organizations and enables effective integration of services and products. For example, a remediation product may use CVE information from several scanning products and monitoring sensors, enabling an integrated risk mitigation solution. CVE provides the following:

- A comprehensive list of publicly known software flaws
- A globally unique name to identify each vulnerability
- A basis for discussing priorities and risks of vulnerabilities
- A way for a user of disparate products and services to integrate vulnerability information

A CVE vulnerability entry consists of a unique name (e.g., CVE-2000-0001), a short description (e.g., “RealMedia server allows remote attackers to cause a denial of service via a long ramgen request.”), and references to public advisories on the vulnerability.

CVE is used in conjunction with other SCAP specifications to satisfy the following use cases:

- **XCCDF.** In an XCCDF checklist, CVEs are used to uniquely identify which software flaw vulnerabilities are of interest (i.e., flaws that are to be checked during the evaluation of the checklist).
- **CVSS.** CVSS scores are associated with CVE entries to uniformly express the fundamental characteristics of the software flaw and to provide a severity score based on these characteristics.
- **OVAL.** Including the specific CVE entry in the OVAL metadata enables a reviewer to accurately understand the basis for a given OVAL Definition such as a Vulnerability or Patch test.

Working with researchers, The MITRE Corporation assigns CVE IDs to publicly known vulnerabilities in commercial and open source software.<sup>18</sup>

### 3.3 Common Vulnerability Scoring System (CVSS) 2.0

CVSS 2.0 provides a repeatable method for consistently evaluating and expressing the risk associated with a given software flaw (e.g., CVE). The use of this shared scoring model allows meaningful comparisons of vulnerability severity scores. CVSS provides three metric groups that can be used to derive a vulnerability score:

- **Base,** which uses the intrinsic characteristics of the vulnerability to provide a generic score
- **Temporal,** which captures external factors that may change over time (e.g., availability of exploit code). The base score is adjusted to render a temporal score that accounts for the temporal factors.
- **Environmental,** which characterizes the severity of a vulnerability in the context of an organization’s operating environment.

---

<sup>17</sup> CVE issuance is managed by The MITRE Corporation and is sponsored by the DHS National Cyber Security Division (NCSA). General CVE information is available at <http://cve.mitre.org/>.

<sup>18</sup> The CVE repository maintained by NIST contains all CVEs issued by The MITRE Corporation as well as supplemental data such as CVSS base scores, vendor statements, and Spanish language translations. NVD provides fine-grained searching and statistical analysis capabilities as well. CVEs and associated NIST-provided metadata can be viewed at <http://nvd.nist.gov/nvd.cfm>.

The purpose of performing CVSS scoring is to help organizations understand the relative importance of various vulnerabilities so that they can effectively assess, prioritize, and mitigate vulnerabilities. Because hundreds of vulnerabilities are publicly announced every week, it is important for organizations to have an easy way to identify those vulnerabilities that have the greatest operational impact. NVD analysts compute and publish CVSS base scores for all CVEs, but organizations are encouraged to further tailor these scores by employing the temporal and environmental metrics to more precisely measure the risk a vulnerability represents within their specific organization.

Complete examples of CVSS measures and scores are available in the official CVSS 2.0 specification [MEL07]. A brief example of base measures, extracted from [MEL07], is [AV:N/AC:L/Au:N/C:C/I:C/A:C], with a base score of 10.0. The bracketed notation for the base measures is known as a *vector*. The first half of the notation indicates that the Access Vector is Network, the Access Complexity is Low, and the Authentication requirement is None. The second half of the notation indicates that the potential impact to Confidentiality, Integrity, and Availability is Complete. The scoring scale is 0 to 10, with 10 being the most severe, so a score of 10.0 indicates the highest severity possible.

The CVSS Special Interest Group (CVSS-SIG) from the Forum of Incident Response and Security Teams (FIRST) developed CVSS 2.0. More information on CVSS can be found at <http://www.first.org/cvss>.

## 4. SCAP General Requirements and Conventions

As described in NIST SP 800-117, *Guide to Adopting and Using the Security Content Automation Protocol*,<sup>19</sup> the motivation for creating SCAP was to provide a standardized approach to maintaining the security of enterprise systems, enhance interoperability of security products, and enable consistent security assessments. The following conventions and requirements were established to help satisfy these goals by ensuring that validated products and content interoperate as designed and provide the expected results.

### 4.1 Support for Legacy SCAP Versions

Products supporting SCAP 1.1 SHALL process SCAP 1.0 content as described under the SP 800-126 for SCAP 1.0.<sup>20</sup>

### 4.2 XCCDF Conventions and Requirements

An SCAP XCCDF document is a machine-readable XML instance document that defines the policies and test conditions to be evaluated. Types of XCCDF documents include *Definition* documents that express policy statements and *Result* documents that contain the outcomes of an SCAP evaluation.

An SCAP XCCDF instance, or XCCDF component stream, SHALL use the `<xccdf: Benchmark>` element as the document element. Each XCCDF component stream SHALL validate against the XCCDF schema (<http://scap.nist.gov/specifications/xccdf/>) and conforms to all relevant content requirements as outlined in the XCCDF Specification [QUI08].

The following general restrictions apply to SCAP XCCDF content:

1. The use of the `@xml:base` attribute SHALL NOT be allowed in SCAP XCCDF content. This attribute is not compatible with the SCAP data stream model as described in section 5.1.
2. An `@lang` attribute for the `<xccdf: Benchmark>` element MUST be provided.
3. If a `@lang` attribute is omitted within the content model, the `@xml:lang` attribute of the nearest ancestor `<xccdf: Benchmark>`, `<xccdf: Value>`, `<xccdf: Group>`, `<xccdf: Rule>` or `<xccdf: Profile>` element SHALL be consulted.

#### 4.2.1 Metadata Elements

XCCDF metadata provides descriptive information about the document. The metadata is used by SCAP products to assist in the selection of the appropriate SCAP data stream, ensure that the most recent or correct version of an XCCDF document is used, and to provide additional information about the document.

The following requirements and conventions apply to the `<xccdf: Benchmark>`, `<xccdf: Profile>`, `<xccdf: Value>`, `<xccdf: Group>`, and `<xccdf: Rule>` elements:

<sup>19</sup> NIST SP 800-117 is available at <http://csrc.nist.gov/publications/drafts/800-117/draft-sp800-117.pdf>.

<sup>20</sup> NIST SP 800-126 is available at <http://csrc.nist.gov/publications/nistpubs/800-126/sp800-126.pdf>



1. One or more instances of the `<xccdf:title>` element SHALL be provided. Each instance MUST contain a text value that indicates the purpose of the containing element delimited by an OPTIONAL `@xml:lang` attribute. If more than one `<xccdf:title>` element is provided then the language attribute SHALL be provided.

One or more instances of the `<xccdf:description>` element SHALL be provided. Each instance MUST contain text values that represent the purpose and intended audience of the containing element delimited by an OPTIONAL language attribute. If more than one `<xccdf:description>` element is provided then the `@xml:lang` attribute SHALL be provided.

2. One or more instances of the `<xccdf:reference>` element MAY be included. These elements SHALL provide a cross reference to additional information, preferably including a URL, to obtain additional information regarding the benchmark.

All remaining OPTIONAL elements in the XCCDF schema MAY be included at the author's discretion unless otherwise noted in this document.

For an example, refer to Appendix C.1, lines 31-35.

## 4.2.2 Use of CPE Names and CPE Language Statements

For all SCAP content, the applicability of XCCDF `<xccdf:Benchmark>`, `<xccdf:Profile>`, `<xccdf:Group>`, and `<xccdf:Rule>` elements to specific IT platforms SHALL be specified using Common Platform Enumeration (CPE) Names or the `<cpe-lang:platform-specification>` element.

If compound CPE Name statements are necessary, a CPE Language `<cpe-lang:platform-specification>` element SHALL be defined as a child of the `<xccdf:Benchmark>` element. The `@id` attribute for each `<cpe-lang:platform>` element declared in this manner MAY be referenced within an `<xccdf:platform>` element with a corresponding `@idref` attribute. Complex platforms MAY be referenced this way within `<xccdf:Benchmark>`, `<xccdf:Profile>`, `<xccdf:Group>`, and `<xccdf:Rule>` elements.

For example:

```
<platform-specification xmlns="http://cpe.mitre.org/language/2.0">
  <platform id="xp_and_acrobat">
    <logical-test operator="AND" negate="false">
      <fact-ref name="cpe:/o:microsoft:windows_xp"/>
      <fact-ref name="cpe:/a:adobe:acrobat:7.0.9"/>
    </logical-test>
  </platform>
</platform-specification>
<platform idref="xp_and_acrobat"/>
```

One or more `<xccdf:platform>` elements MAY be specified within the `<xccdf:Benchmark>`, `<xccdf:Profile>`, `<xccdf:Group>`, or `<xccdf:Rule>` elements to more narrowly define the applicability of these specific elements. The `<xccdf:platform>` `@idref` attribute SHALL reference either:

1. A CPE Name; or

2. The *@id* attribute of a *<cpe-lang:platform-specification/cpe-lang:platform>* element.

Within a given *<xccdf:Benchmark>*, *<xccdf:Profile>*, *<xccdf:Group>*, or *<xccdf:Rule>* context, if no *<xccdf:platform>* element is defined, the element SHALL be considered to apply to the *<xccdf:platform>* of its nearest ancestor. If none is present, the element SHALL be considered to apply to all applicable targets.

CPE Names used within an XCCDF document SHALL match the names of existing Official CPE Dictionary<sup>21</sup> entries where names for the desired platform exist. If multiple matches are found within the dictionary (e.g., deprecated and current CPE Names), the most current CPE Name SHOULD be used.

Each reference to a CPE Name SHALL be declared in the required CPE dictionary stream and each OVAL inventory class definition referenced from the dictionary stream SHALL be specified in the required CPE inventory stream.

#### 4.2.3 The *<xccdf:Benchmark>* Element

The following requirements and conventions apply to the *<xccdf:Benchmark>* element:

1. The REQUIRED *@id* attribute SHALL be used to uniquely identify all revisions of a benchmark.
2. The *@style* attribute SHALL have the value “SCAP\_1.1”.
3. The *<xccdf:status>* element indicates the current status of the benchmark document. The associated text value MUST be “draft” for documents released in public draft state and “accepted” for documents that have been officially released by an organization. The *@date* attribute SHALL be populated with the date of the status change. Additional *<xccdf:status>* elements MAY be included to indicate historic status transitions.
4. The *<xccdf:version>* element SHALL uniquely identify the particular revision of the benchmark.
5. One or more instances of the *<xccdf:notice>* element MAY be provided indicating clarifications, suggestions, or warnings regarding the use of the benchmark, including but not limited to terms of use, legal notices, or copyright statements.
6. The *<xccdf:metadata>* element SHALL be provided and SHALL, at minimum, contain the Dublin Core<sup>22</sup> terms from Table 4-1. Additional Dublin Core terms SHALL follow the required terms within the element sequence.

---

<sup>21</sup> The Official CPE Dictionary is located at <http://nvd.nist.gov/cpe.cfm>.

<sup>22</sup> <http://dublincore.org/documents/dces/>

**Table 4-1. Use of Dublin Core Terms in XCCDF Metadata**

Dublin Core Term	Description of Use
<code>&lt;dc:creator&gt;</code>	The person, organization, and/or service that created the XCCDF XML instance
<code>&lt;dc:publisher&gt;</code>	The person, organization, and/or service that published the XCCDF XML instance
<code>&lt;dc:contributor&gt;</code>	The person, organization, and/or service that contributed to the creation of the XCCDF XML instance
<code>&lt;dc:source&gt;</code>	An identifier that indicates the organizational context of the <code>&lt;xccdf:Benchmark&gt;</code> element's <code>@id</code> attribute. An organizationally specific URI SHOULD be used.

#### 4.2.4 The `<xccdf:Profile>` Element

The use of an `<xccdf:Profile>` element SHALL NOT be required. If an `<xccdf:Profile>` element is not provided or selected, then profile processing SHALL be skipped and standard XCCDF benchmark processing rules SHALL apply.<sup>23</sup>

#### 4.2.5 The `<xccdf:Rule>` Element

While the previous version of SCAP required the inclusion of CVSS data, NIST currently provides a CVE data feed to support dynamic and current vulnerability information and associated metadata (e.g., CVSS values). The current schema is available at <http://nvd.nist.gov/download.cfm>. This data feed SHOULD be used in place of static CVSS scores in the `@weight` attribute.

##### 4.2.5.1 Embedded CCE References

XCCDF `<xccdf:Rule>` elements MAY be used to define a policy requiring compliance with a specific configuration setting. When a configuration setting having one or more associated CCE Identifiers from the CCE List is expressed as an XCCDF rule, an `<xccdf:ident>` element<sup>24</sup> reference SHALL be provided within the `<Rule>` element. The `<xccdf:ident>` element provides a globally unique identifier for a specific configuration setting.

The `<xccdf:ident>` element syntax SHALL be used to reference CCE Identifiers as follows:

1. The system attribute for the `<xccdf:ident>` element SHALL be defined using the CCE Version 5 system identifiers, either “CCE” or “<http://cce.mitre.org>” (preferred method).
2. The *CCE Identifier* SHALL be used for the `<xccdf:ident>` element content.
3. An `<xccdf:ident>` element referencing a *CCE Identifier* SHALL be ordered before other `<xccdf:ident>` elements referencing non-SCAP identifiers.

<sup>23</sup> See NIST IR 7275r3, The XCCDF Specification version 1.1.4, p.36 section “Benchmark Processing Algorithm” for additional details.

<sup>24</sup> See NIST IR 7275r3, The XCCDF Specification version 1.1.4, p.21 table, p.22 paragraph 5, and p.59 section “<ident>” for additional details.

For example:

```
<Rule id="AuditAccountLogonEvents">
  <title>Audit Account Logon Events</title>
  ...
  <ident system="http://cce.mitre.org">CCE-3867-0</ident>
  <ident system="http://cce.mitre.org">CCE-3008-0</ident>
  ...
</Rule>
```

#### 4.2.5.2 Embedded CVE References

XCCDF `<xccdf:Rule>` elements MAY be used to assess security related software flaws. When this assessment is associated with one or more associated *CVE Identifiers* from the *CVE vulnerability feeds*, an `<xccdf:ident>` element reference within the `<xccdf:Rule>` element SHALL be provided.

The `<xccdf:ident>` element syntax SHALL be used as follows:

1. The system attribute for the `<xccdf:ident>` element SHALL be defined using the CVE system identifiers, either “CVE” or “<http://cve.mitre.org>” (preferred method).
2. The *CVE Identifier* SHALL be used for the `<xccdf:ident>` element content.
3. An `<xccdf:ident>` element referencing a *CVE Identifier* SHALL be ordered before other `<xccdf:ident>` elements referencing non-SCAP identifiers.

For example:

```
<Rule id="SQLInjectionVulnerability"
  <title>SQL Injection Vulnerability</title>
  ...
  <ident system="http://cve.mitre.org">CVE-2008-6865</ident>
  <ident system="http://cve.mitre.org">CVE-2008-6866</ident>
  ...
</Rule>
```

#### 4.2.5.3 Use of a Patches Up-To-Date Rule

An OVAL instance document may be used to represent a series of checks to verify that patches have been installed. Historically, an XCCDF convention has been used to identify such a reference. When implementing a patches up-to-date XCCDF rule, the following approach SHALL be used:

1. The `<xccdf:Rule>` element that references an OVAL Patch component SHALL have the `@id` attribute value of “*patches\_up\_to\_date*”.
2. A single `<xccdf:check>` element SHALL be provided for the `<xccdf:Rule>` with a `@system` attribute value of “*http://oval.mitre.org/XMLSchema/oval-definitions-5*”.

3. Each `<xccdf:check-content-ref>` element SHALL have an `@href` attribute referencing a valid SCAP `<oval-def:oval_definitions>` document instance (see Section 4.3.3) with the `@name` attribute omitted.
4. Each OVAL definition with a `@class` attribute defined as “patch” SHALL represent a check for the presence of a discrete patch.

For example:

```
<Rule id="patches_up_to_date" selected="false">
  <title>Security Patches Up-To-Date</title>
  <description>Keep systems up to current patch levels</description>
  <check system="http://oval.mitre.org/XMLSchema/oval-definitions-5">
    <check-content-ref href="scap-win2000-patches.xml" />
  </check>
</Rule>
```

## 4.2.6 Allowed Check System Usage

All XCCDF requirements and conventions apply to the use of the `<xccdf:check>` and `<xccdf:complex-check>` elements relative to valid SCAP content and scanning products with the following restrictions:

1. The `<xccdf:check-content>` element SHALL NOT be used to embed check content directly into XCCDF content.
2. At least one `<xccdf:check-content-ref>` element MUST be provided.
3. If multiple `<xccdf:check-content-ref>` elements are provided, then the following evaluation method SHALL be performed:
  - a. Evaluate each `<xccdf:check-content-ref>` element in the order that it appears in the `<xccdf:check>` element. The first resolvable `<xccdf:check-content-ref>` element SHALL be used to determine the `<xccdf:Rule>` status.
  - b. For each `<xccdf:check-content-ref>` element, a product will attempt to retrieve the document referenced by the `@href` attribute. If not resolvable, the next available `<xccdf:check-content-ref>` element SHALL be evaluated. If none of the `<xccdf:check-content-ref>` elements are resolvable, then the result of the rule evaluation SHALL be the XCCDF “error” status and processing of the `<xccdf:Rule>` SHALL end. Please note that it is acceptable to map a remote URL to a local copy of the file in cases where remote access is not available, not allowed, or not practical.
  - c. Once a resolvable `<xccdf:check-content-ref>` element is found, then check system processing SHALL proceed. When evaluating a rule, an `<xccdf:rule-result/xccdf:message>` with the `@severity` attribute value of ‘info’ SHALL be generated, indicating the `<xccdf:check-content-ref>` href and name, if provided.”

For example, refer to Appendix C.1 lines 223-226.

4. Use of XCCDF check systems as specified in the `<xccdf:check>` element's `@system` attribute SHALL be restricted as follows:
  - a. The following check systems SHALL be *supported* by SCAP:
    - i. Use of the OVAL check system SHALL be indicated by the `http://oval.mitre.org/XMLSchema/oval-definitions-5` system identifier. See Section 4.2.6.1.
    - ii. Use of the OCIL check system SHALL be indicated by the `http://www.mitre.org/ocil/2` system identifier. See Section 4.2.6.2.
  - b. SCAP content SHALL be considered *well-formed* if an SCAP *supported* check system is used.
  - c. If a check system is used in XCCDF content that is not *supported* by SCAP, then this content SHALL NOT be considered *well-formed* with regards to SCAP.
  - d. SCAP scanning products SHALL *implement* the SCAP *supported* check systems.
  - e. SCAP scanning tools MAY *implement* non-SCAP check systems that are not *supported* by SCAP.
  - f. Evaluation of an `<xccdf:check>` containing a reference to a non-SCAP check system SHALL produce an "unknown" result if an SCAP scanning product does not *implement* the check system.

#### 4.2.6.1 Use of the OVAL as a Check System

An `<xccdf:Rule>` MAY refer to one or more OVAL Definitions to implement the technical tests necessary to determine the pass/fail status of the rule.

##### 4.2.6.1.1 OVAL `<xccdf:check>` Usage

References from SCAP compliant XCCDF to OVAL Definitions SHALL use the form:

```
<check-content-ref href="OVAL_Source_URI" [name="OVAL_Definition_Id"] />
```

The `@href` attribute SHALL reference an OVAL component stream which is described in Section 4.3.3. When present, the OPTIONAL `@name` attribute SHALL refer to a specific OVAL Definition in the designated component stream.

In the previous example, the `<xccdf:check-content-ref>` element's `@href` attribute refers to an OVAL Definitions stream containing one or more OVAL patch definitions. This `<xccdf:check-content-ref>` is equivalent to *referencing* a virtual OVAL Definition of the form:

```
<oval_definitions xmlns:oval-def="http://oval.mitre.org/XMLSchema/oval-definitions-5">
  <definitions>
    <definition id="identifier of patch definition" version="0" class="patch">
      ...
    <criteria >
```

```

        <extend_definition definition_ref="identifier of patch definition 1" />
        ...
        <extend_definition definition_ref="identifier of patch definition N" />
    </criteria>
</definition>
</definitions>
</oval_definitions>

```

where the extended definitions are the individual patch definitions defined in the OVAL content stream.<sup>25</sup>

#### 4.2.6.1.2 <xccdf:value> and OVAL Variable Dependencies

One or more <xccdf:check-export> elements MAY be used to define the binding of <xccdf:value> elements to OVAL variables. The format of the <xccdf:check-export> element is:

```

<check-export xmlns="http://checklists.nist.gov/xccdf/1.1"
    value-id="XCCDF_Value_id" export-name="OVAL_External_Variable_id" />

```

The following check element example demonstrates the use of this convention:

```

<check system="http://oval.mitre.org/XMLSchema/oval-definitions-5">
    <check-export export-name="oval:gov.nist.fdcc.xp:var:66711"
        value-id="NoSlowLink_var" />
    <check-export export-name="oval:gov.nist.fdcc.xp:var:66712"
        value-id="NoBackgroundPolicy_var" />
    <check-export export-name="oval:gov.nist.fdcc.xp:var:66713"
        value-id="NoGPOListChanges_var" />
    <check-content-ref href="fdcc-winxp-oval.xml"
        name="oval:gov.nist.fdcc.xp:def:6671" />
</check>

```

The type and value binding of the specified XCCDF Value is constrained to match that lexical representation of the indicated OVAL Variable Data Type. Table 4-2 summarizes the constraints regarding data type usage. Additional information regarding OVAL and XCCDF data types can be found in the OVAL Common Schema documentation<sup>26</sup> and the XCCDF specification<sup>27</sup>.

**Table 4-2. XCCDF-OVAL Data Export Matching Constraints**

OVAL Data Type	Matching XCCDF Data Type
int	number
float	number
boolean	boolean

<sup>25</sup> The Inventory definition results are not to be considered in the overall patch results. The intent is to logically combine the result of each definition with a class of 'patch' using the AND operation. The patch definition file may contain a mix of patch and inventory definitions and the overall patch checking result should be the AND of just the patch definition results values.

<sup>26</sup> <http://oval.mitre.org/language/download/schema/version5.4/ovaldefinition/documentation/oval-common-schema.html#DatatypeEnumeration> and

<sup>27</sup> <http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/documentation/oval-definitions-schema.pdf>  
<http://csrc.nist.gov/publications/nistir/ir7275r3/NISTIR-7275r3.pdf>

OVAL Data Type	Matching XCCDF Data Type
string, evr_string, version, ios_version, fileset_revision, binary	string

#### 4.2.6.1.3 Mapping OVAL Results to XCCDF Results

When evaluating an `<xccdf:Rule>` element that references an OVAL Definition, the `<xccdf:rule-result>` element is used to capture the result of this evaluation. This result is determined by evaluating the referenced OVAL Definition on a target host. The `<xccdf:result>` value recorded is mapped from the OVAL Definition Result produced during evaluation.

In some cases the derived results may seem counterintuitive, but when viewed in the appropriate context the underlying logic is evident. For example, if an OVAL Definition of class “compliance” is processed and the XCCDF returns a result of “True”, the product is conveying the fact that the system was found to be compliant with that check and therefore returns a “Pass” result. A similar definition for a vulnerable condition will return results of “False” if that vulnerability was not found on the examined devices, resulting in a “Pass” from the XCCDF rule.

SCAP compliant processors that generate XCCDF `<xccdf:rule-result>` elements SHALL apply the mapping illustrated in Table 4-3 when deriving `<xccdf:Rule>` results from OVAL Definition processing. The corresponding `<xccdf:rule-result/xccdf:result>` value SHALL be recorded based on the `@class` of the OVAL Definition where applicable.

**Table 4-3. Deriving XCCDF Rule Results from OVAL Definition Results**

OVAL Definition Result		XCCDF Rule Result
error		error
unknown		unknown
not applicable		notapplicable
not evaluated		notchecked
Definition Class	Definition Result	pass
compliance	true	
vulnerability	false	
inventory	true	
patch	false	
Definition Class	Definition Result	fail
compliance	false	
vulnerability	true	
inventory	false	
patch	true	

#### 4.2.6.2 Use of OCIL as a Check System

Similar to OVAL, an `<xccdf:Rule>` MAY refer to one or more OCIL Questionnaires for tests that requires manual processing (e.g., requires human feedback) to determine the result of the rule.

##### 4.2.6.2.1 OCIL `<xccdf:check>` Usage

References from SCAP compliant XCCDF to OCIL Questionnaires SHALL use the form:



```
<check-content-ref href="OCIL_Source_URI" name="OCIL_Questionnaire_Id" />
```

The *@href* attribute SHALL identify the OCIL Questionnaire XML stream. The *@name* attribute SHALL refer to a specific OCIL Questionnaire in the designated content stream. When an XCCDF rule references a specific OCIL Questionnaire, an OCIL Questionnaire source SHALL be available to resolve the reference.

In the previous example, the `<xccdf:check-content-ref>` element's *@href* attribute refers to an OCIL Questionnaire stream containing one or more test actions. This *@check-content-ref* is equivalent to *referencing* a virtual OCIL Questionnaire of the form:

```
<questionnaire xmlns:ocil="http://www.mitre.org/ocil/2" id="identifier of questionnaire"
scope="FULL" priority="MEDIUM" child_only="false">
  <title>...</title>
  <description>...</description>
  <references>...</references>
  <actions>
    <test_action_ref>identifier of test action</test_action_ref>
  </actions>
  ...
</questionnaire>
```

#### 4.2.6.2.2 `<xccdf:Value>` and OCIL Variable Dependencies

OCIL provides support for defining values that can be reused in stating questions and evaluating questionnaires. These values MAY be passed from XCCDF as value parameters and into OCIL as external variables.

When an XCCDF Rule refers to an OCIL Questionnaire requiring one or more external variables, it SHALL include `<xccdf:check-export>` elements that define bindings from XCCDF values to OCIL variables. All check-export elements SHALL precede the check-content-ref element.

The format of these elements is:

```
<check-export xmlns="http://checklists.nist.gov/xccdf/1.1"
value-id="XCCDF_Value_id" export-name="OCIL_External_Variable_id" />
```

The following check element example demonstrates the use of this convention:

```
<check system="http://www.mitre.org/ocil/2">
  <check-export export-name="ocil:gov.nist.fdcc.xp:var:11000"
value-id="OnlyLaptops_var" />
  <check-export export-name="ocil:gov.nist.fdcc.xp:var:11001"
value-id="HighAlertPolicy_var" />
  <check-content-ref href="fdcc-all-ocil.xml"
name="ocil:gov.nist.fdcc.xp:def:11001" />
</check>
```

The type and value binding of the specified XCCDF Value SHALL be constrained to match that lexical representation of the indicated OCIL Variable Data Type. Table 4-4 summarizes the constraints regarding data type usage.

**Table 4-4. XCCDF-OCIL Data Export Matching Constraints**

OCIL Variable Data Type	Matching XCCDF Data Type
TEXT	string; boolean
NUMERIC	number

#### 4.2.6.2.3 Mapping OCIL Results to XCCDF Results

Similar to OVAL, OCIL supports full status reporting that includes Error, Unknown, Not Tested, Not Applicable, Pass, and Fail. See Table 4-5 for mapping from OCIL Questionnaire Results to XCCDF Results. When evaluated, an OCIL document MAY contain an `<inter:results>` element that contains all the questionnaire results, including user responses and artifacts.

**Table 4-5. Deriving XCCDF Rule Results from OCIL Questionnaire Results**

OCIL Questionnaire Result	XCCDF Rule Result
Error	error
Unknown	unknown
Not Applicable	notapplicable
Not Tested	notchecked
Pass	pass
Fail	fail

#### 4.2.7 XCCDF Test Results

XCCDF test results are documented as the contents of an `<xccdf:TestResult>` element that either stands alone as the root of an XML document or is embedded as a child-element of an `<xccdf:Benchmark>` root element. In the latter case, the associated benchmark is the embedding benchmark; in the former, the `<xccdf:TestResults>` document requires an embedded `<xccdf:Benchmark>` element that identifies the associated benchmark. `<xccdf:Benchmark>` elements are ignored in `<xccdf:TestResult>` elements that are embedded in their associated benchmark.

To be considered valid SCAP result content, the following conditions SHALL be met:

1. One or more `<xccdf:organization>` elements SHALL be provided to indicate the organizational units responsible for applying the checklist.
2. The `@start-time` and `@end-time` attributes SHALL be provided to indicate when the scan started and completed, respectively.
3. The `@test-system` attribute SHALL be provided with a CPE Name value indicating the product that evaluated the checklist.
4. Regarding the definition and use of `<xccdf:Profile>` elements:

- a. If no `<xccdf:Profile>` was selected, then the `<xccdf:Profile>` SHALL be omitted.
  - b. When using a profile during the processing of XCCDF content, the test results SHALL embed an `<xccdf:Profile>` element that identifies the non-abstract profile in the associated benchmark whose evaluation results are reported by the test results.
  - c. Reported rule results SHALL include all selected rules within the specified Profile.
  - d. Reported value-settings SHALL include all those values that are exported by the reported rules. The specific settings are those determined by the reported Profile.
  - e. The `<identity>` tag SHALL identify the security principal used to access rule evaluation on the target(s).
5. Each IP address associated with the `<xccdf:target>` SHALL be enumerated using the `<xccdf:target-address>` element.
  6. The `<rule-result>` elements SHALL report the result of the application of each selected rule against all specified targets. The `@idref` attribute of the `<xccdf:rule-result>` SHALL identify the selected rule and each `<xccdf:instance>` element SHALL identify the corresponding `<xccdf:target>` element.
  7. Where applicable to the target system, each of the following `<xccdf:fact>` elements SHALL be provided:

**Table 4-6. XCCDF Fact Descriptions**

XCCDF Fact	Description of Use
<code>urn:scap:fact:asset:identifier:mac</code>	Ethernet media access control address
<code>urn:scap:fact:asset:identifier:ipv4</code>	Internet Protocol version 4 address
<code>urn:scap:fact:asset:identifier:ipv6</code>	Internet Protocol version 6 address
<code>urn:scap:fact:asset:identifier:host_name</code>	Host name of the asset, if assigned
<code>urn:scap:fact:asset:identifier:fqdn</code>	Fully qualified domain name
<code>urn:scap:fact:asset:identifier:ein</code>	Equipment identification number or other inventory tag number
<code>urn:scap:fact:asset:identifier:guid</code>	Globally unique identifier for the asset, if assigned
<code>urn:scap:fact:asset:environmental_information:owning_organization</code>	Organization that tracks the asset on its inventory
<code>urn:scap:fact:asset:environmental_information:current_region</code>	Geographic region where the asset is located
<code>urn:scap:fact:asset:environmental_information:administration_unit</code>	Name of the organization that does system administration for the asset

#### 4.2.7.1 Assigning CVE Identifiers to Rule Results

The XCCDF `<xccdf:rule-result>` element provides data indicating the result of assessing a system using the identified XCCDF `<xccdf:Rule>` element. If the target XCCDF `<xccdf:Rule>` identified by the `<xccdf:rule-result idref="">` attribute has one or more `<ident>`

elements<sup>28</sup> with the “<http://cve.mitre.org>” system identifier, then each `<xccdf:ident>` element SHALL also appear within the `<xccdf:rule-result>` element.

For example:

```
<rule-result idref="java-upgrade-278"
  xmlns=http://checklists.nist.gov/xccdf/1.1 weight="10.0">
  <result>pass</result>
  ...
  <ident system="http://cve.mitre.org">CVE-2006-0614</ident>
  ...
</rule-result>
```

An `<xccdf:rule-result>` of “pass” SHALL indicate that the target platform satisfies all the conditions of the XCCDF rule and is unaffected by the vulnerability or exposure referenced by the CVE.

#### 4.2.7.2 Assigning CCE Identifiers to Rule Results

The XCCDF `<xccdf:rule-result>` element provides data indicating the result of assessing a system using the identified XCCDF `<xccdf:Rule>` element. If the target XCCDF `<xccdf:Rule>` identified by the `<xccdf:rule-result>` `@idref` attribute has one or more `<xccdf:ident>` elements with the “<http://cce.mitre.org>” system identifier, then each `<xccdf:ident>` element SHALL also appear within the `<rule-result>` element. For example:

```
<rule-result idref="minimum_password_length"
  xmlns="http://checklists.nist.gov/xccdf/1.1"
  <result>pass</result>
  ...
  <ident system="http://cce.mitre.org">CCE-2981-9</ident>
  ...
</rule-result>
```

An `<xccdf:rule-result>` of “pass” SHALL indicate that the target platform complies with the configuration setting guidance expressed in the XCCDF rule.

### 4.3 OVAL Conventions and Requirements

When used for SCAP purposes, OVAL content SHALL comply with one of the following document schema:

- `<oval-def:oval_definitions>` document – A specification of OVAL Definitions, Tests, Objects, States, and Variables. This document MAY be used as a component of an SCAP data source.
- `<oval-var:oval_variables>` document – A specification of external OVAL Variable bindings. Instances of this schema MAY be used as a component of an SCAP data stream to pass variable values to an OVAL interpreter.
- `<oval-sc:oval_system_characteristics>` document – A specification of target system characteristics, that is, the specification of OVAL Object values queried from a target system.

<sup>28</sup> See NIST IR 7275r3, The XCCDF Specification version 1.1.4, p.30 table and p.59 section “<ident>”.

- `<oval-res:oval_results>` document – The evaluation results of specified definitions and tests, as well as a copy of the OVAL System Characteristics from which the results can be derived.

#### 4.3.1 Supported Previous Versions of OVAL (5.3, 5.4, and 5.5)

While the default version<sup>29</sup> of OVAL used in SCAP SHALL be OVAL version 5.6, content authors SHOULD utilize the earliest SCAP supported version of OVAL that includes all required tests. Products SHALL support OVAL Definition documents written against OVAL versions 5.3, 5.4, 5.5 and 5.6. This approach, often referred to as the “least-version-principle”, allows for SCAP content to remain viable over a longer period of time.

#### 4.3.2 Support for Deprecated Constructs in OVAL

Within the OVAL Language, constructs may be deprecated.<sup>30</sup> Deprecated constructs MUST be handled properly during OVAL Definition evaluation. Similar to the requirement to support previous minor versions of OVAL, this requirement will ensure that content that made use of these deprecated constructs continues to be supported in SCAP.

#### 4.3.3 OVAL Schema Specification

An SCAP OVAL instance, or OVAL component stream, SHALL use the `<oval-def:oval_definitions>` element as the document element. Each OVAL component stream SHALL validate against the OVAL schema.

All of the OVAL content MUST contain an `<oval:generator>` element. The version of any particular document instance SHALL be specified using the `<oval:schema_version>` content element of the `<oval:generator>` as in this example:

```
<oval:generator>
  <oval:product_name>The OVAL Repository</oval:product_name>
  <oval:schema_version>5.6</oval:schema_version>
</oval:generator>
```

The version of an `<oval-def:oval_definitions>` document SHALL be determined based on the version provided in the `<oval:generator>` element's `<oval:schema_version>` element.

The version of an `<oval-var:oval_variables>` document SHALL be the same as that of the `<oval-def:oval_definitions>` document whose external variables are bound by the variables document.

#### 4.3.4 OVAL Results

An SCAP OVAL Results instance, or OVAL Result component stream, SHALL use the `<oval-res:oval_results>` element as the document element. Each OVAL Result component stream SHALL validate against the OVAL Results schema.

OVAL Result component streams MUST be compliant with version 5.6 of the OVAL Results schema<sup>31</sup>

<sup>29</sup> The OVAL Language versioning methodology is available here: <http://oval.mitre.org/language/about/versioning.html>

<sup>30</sup> The OVAL Language Deprecation policy is available here: <http://oval.mitre.org/language/about/deprecation.html>

<sup>31</sup> The OVAL schemas are described in detail at <http://oval.mitre.org/language/about>.

regardless of the version of the OVAL Definitions document that was evaluated. In order to be SCAP compliant, an SCAP scanning product SHALL be able to produce both thin and full OVAL Results output as described below. The specific result output SHALL be configurable within the SCAP product.

While the OVAL specification permits limiting result status reporting, SCAP-compliant content includes full status reporting including Error, Unknown, Not Applicable, Not Evaluated, True, and False. Section 4.2.6.1.3 provides additional detail about OVAL Results as they relate to XCCDF.

In order to support SCAP instances where OVAL thin content (only the ID of the definition and the results) is preferred, SCAP products SHALL support all valid values for the `<oval-res:directives>` controlling the expected content of the results file.

To support the ability for results to be consumed by the appropriate product(s), data results SHALL be expressed as Single Machine Without System Characteristics, Single Machine With System Characteristics, or Single Machine with Thin Results as follows:

1. Single Machine Without System Characteristics – A single result file that includes all OVAL definitions evaluated and “full” results types as described in the ContentEnumeration element of the OVAL Results schema<sup>32</sup>, without system characteristics.

For this format, the values for the `<oval-res:directives>` element SHALL be:

```
<definition_true content="full" reported="true"/>
<definition_false content="full" reported="true"/>
<definition_unknown content="full" reported="true"/>
<definition_error content="full" reported="true"/>
<definition_not_evaluated content="full" reported="true"/>
<definition_not_applicable content="full" reported="true"/>
```

2. Single Machine With System Characteristics – A single result file that includes all OVAL definitions evaluated and “full” results types as described in the ContentEnumeration element of the OVAL Results schema and the System Characteristics of the target evaluated.

For this format, the values for the `<oval-res:directives>` element SHALL be:

```
<definition_true content="full" reported="true"/>
<definition_false content="full" reported="true"/>
<definition_unknown content="full" reported="true"/>
<definition_error content="full" reported="true"/>
<definition_not_evaluated content="full" reported="true"/>
<definition_not_applicable content="full" reported="true"/>
```

3. Single Machine with Thin Results – A single result file that includes all OVAL definitions evaluated and “thin” results types as described in the OVAL Results schema. A value of 'thin' means only the minimal amount of information will be provided.

For this format, the values for the `<oval-res:directives>` element SHALL be:

```
<definition_true content="thin" reported="true"/>
<definition_false content="thin" reported="true"/>
<definition_unknown content="thin" reported="true"/>
<definition_error content="thin" reported="true"/>
```

---

<sup>32</sup> The OVAL Results schema is described at <http://oval.mitre.org/language/about/structure.html#results>.

```
<definition_not_evaluated content="thin" reported="true"/>
<definition_not_applicable content="thin" reported="true"/>
```

#### 4.4 OCIL Conventions

OCIL content SHALL be used for checking rules that cannot be fully automated with OVAL or that require human input. An SCAP OCIL instance, or OCIL component stream, SHALL use the `<ocil:ocil>` element as the document element. Each OCIL component stream SHALL validate against the OCIL schema. Similar to OVAL, all content MUST contain an `<inter:generator>` element describing the creation of the document. The version of any OCIL document instance SHALL be specified using the `<inter:schema_version>` element.

For example:

```
<inter:generator>
  <author>
    <name>John Doe</name>
    ...
  </author>
  <inter:schema_version>2.0</inter:schema_version>
  <inter:timestamp> 2009-08-24T12:06:31</inter:timestamp>
  <inter:product_name>OCIL Document</inter:product_name>
</inter:generator>
```

The version used to validate an OCIL document instance MUST be chosen based on the version provided by the `<inter:generator>`/`<inter:schema_version>` element's value.

An OCIL document MUST contain at least one `<inter:questionnaire>` element, one `<inter:test_action>` derived element, and one `<inter:question>` derived element that describes a single check. It MAY contain `<inter:variables>` for specifying values: 1) defined by the author, 2) derived based on answers to questions, or 3) declared externally. It MAY also contain `<inter:artifacts>` to hold evidence or supporting objects for user responses to questions.

#### 4.5 CPE Conventions

The Official CPE Dictionary data feed<sup>33</sup> may be used by SCAP components to reference CPE Names. Local enumerations are permitted, but if a CPE Name for a product or platform exists in the Official CPE Dictionary, the content SHALL match the product or platform referenced by that official identifier.

Section 8 of CPE Specification 2.2 provides the defining structure of the Official CPE Dictionary. For certain names, a `<cpe_dict:cpe-item>` MAY contain one or more `<check>` elements that references OVAL system inventory definitions using the following format:

```
<cpe_dict:check system="http://oval.mitre.org/XMLSchema/oval-definitions-5"
  href="Oval_URL">Oval_inventory_definition_id</cpe_dict:check>
```

For example:

```
<cpe-list xmlns="http://cpe.mitre.org/dictionary/2.0"
  xmlns:cpe_dict="http://cpe.mitre.org/dictionary/2.0">
```

<sup>33</sup> The Official CPE Dictionary is located at <http://nvd.nist.gov/cpe.cfm>.

```
<cpe-item name="cpe:/o:microsoft:windows_2003">
  <title>Microsoft Windows Server 2003</title>
  <check system="http://oval.mitre.org/XMLSchema/oval-definitions-5">
    oval:org.mitre.oval:def:128
  </check>
</cpe-item>
</cpe-list>
```

The referenced OVAL inventory definition specifies the technical procedure for determining whether or not a specific target asset is an instance of the CPE Name specified by the `<cpe_dict:cpe-item>` element. This usage is encouraged for a CPE dictionary component stream that is part of an SCAP expressed data stream.

If a `<cpe_dict:cpe-item>` contained in a CPE dictionary component of an SCAP data stream references an OVAL “inventory” definition, then that definition SHALL be resolved by a CPE Inventory component stream in the same data stream.<sup>34</sup> Furthermore, the title of the `<cpe_dict:cpe-item>` SHALL match the title of an affected platform bound to the referenced definition.<sup>35</sup>

## 4.6 CCE Conventions

CCE identifiers are used by SCAP components to reference Common Configuration Enumerations. CCE identifiers for new configuration settings are assigned by the CCE Content Team.<sup>36</sup> To maintain consistency and accuracy among the SCAP validated products, if a CCE entry for a particular configuration setting exists in the Official CCE Dictionary, the security products SHALL use the official CCE identifier. If no CCE exists for the configuration setting of interest, an alternate identifier MAY be used, but the user SHOULD seek to have a CCE identifier issued for the configuration setting.

The MITRE Corporation maintains the current official CCE list at [http://cce.mitre.org/lists/cce\\_list.html](http://cce.mitre.org/lists/cce_list.html) and new CCEs can be requested from The MITRE Corporation at [http://cce.mitre.org/lists/creation\\_process.html](http://cce.mitre.org/lists/creation_process.html).

Use of an official, dynamic data feed is preferred to static coding of values in SCAP data sources. The NVD provides a data feed<sup>37</sup> that correlates CCE identifiers with the control identifiers described in SP 800-53.

## 4.7 CVE Conventions

CVE identifiers are used by SCAP components to reference publicly known software flaws.<sup>38</sup> CVE references in SCAP content MAY include both “candidate” and “entry” status identifiers. The use of deprecated CVE identifiers SHALL NOT be allowed. If a CVE identifier exists for a particular vulnerability, security products SHALL use the official CVE identifier. If no CVE exists for the software flaw, an alternate identifier MAY be used, but the user SHOULD seek to have a CVE identifier issued for the vulnerability. The process for submitting unpublished vulnerabilities and obtaining CVE identifiers is available from The MITRE Corporation via [http://cve.mitre.org/cve/obtain\\_id.html](http://cve.mitre.org/cve/obtain_id.html).

<sup>34</sup> More information is provided in Section 4.7.

<sup>35</sup> Section 4.2.2 explains more detail about OVAL Definitions.

<sup>36</sup> [http://cce.mitre.org/lists/creation\\_process.html](http://cce.mitre.org/lists/creation_process.html) documents the CCE Creation Process.

<sup>37</sup> <http://web.nvd.nist.gov/view/ncp/checklist-cce-feed?id=113&cid=2>

<sup>38</sup> NIST provides the NVD CVE data feed at [http://nvd.nist.gov/download.cfm#CVE\\_FEED](http://nvd.nist.gov/download.cfm#CVE_FEED).



It should be noted that not all CVE entries identify an associated patch or remediation; in fact, the ability to determine the availability of a patch or remediation is a valuable feature of the CVE component. Vendors **SHOULD** reference CVE entries in notifications (e.g., security patch bulletins) to support the use of automated products and to ensure clarity when referencing a given vulnerability. Similarly, CVE authors **SHOULD** reference applicable vendor patch identification whenever possible.

NIST provides a CVE data feed to support dynamic and current vulnerability information and associated metadata (e.g., CVSS values). The current schema is available at <http://nvd.nist.gov/download.cfm>.

## **4.8 CVSS Conventions**

The CVSS specification (described at <http://www.first.org/cvss/cvss-guide.html>) defines Base score metrics that characterize the severity of the vulnerability using the intrinsic characteristics of the vulnerability. The CVSS framework also allows further refinement of the base score using Temporal Metrics, which may change over time (e.g., Exploitability, Report Confidence) and Environmental Metrics, which are unique to a particular environment (e.g., Collateral Damage Potential, Target Distribution).

If an accompanying CVSS score exists for a CVE, products **SHOULD** include CVSS scores and vector strings in CVE-related output produced by the product. CVSS base scores and vector data are provided for all CVE identifiers contained in the NVD CVE data feed. If a CVSS Base Metric is provided, it **SHALL** reflect the current Base score as reflected in the official source. SCAP users **MAY** leverage the flexibility provided within the CVSS component specification by deriving and using the Temporal and Environmental metrics as needed.

## 5. SCAP Use Case Requirements

To facilitate implementation of the SCAP requirements specified in Section 4, this section describes specific uses that demonstrate effective use of the protocol. The content of this section identifies the input data source conventions identified with the SCAP components and the resulting data produced. These data streams are associated with the following use case examples:

- Configuration Verification
- Vulnerability Assessment
- Patch Validation
- Inventory Collection

These examples are not intended to limit SCAP, but to provide a framework for future use cases and document the specifics of the data streams described. SCAP enables many types of automated assessment, each with discrete benefits and each considered separate content. For example, vulnerability assessment (i.e., quantitative and repeatable measurement and scoring of software flaw vulnerabilities across systems) is related to, but separate from, configuration verification.

### 5.1 SCAP Data Streams

An SCAP data stream is a collection of XML instance documents, also called “*component streams*”. An instance of an SCAP data stream is the expression of a specific use case using one or more SCAP component streams that can be processed by an SCAP product. The required XML content composing an SCAP data stream depends on the use case and is designed to satisfy specific policy or situational awareness objectives.

#### 5.1.1 SCAP Source Data Streams

Every SCAP data stream bundle SHALL use a common locator prefix that is part of a relative URL whose base is the URL of the deployed data source. The notation ‘xxx-’ designates a locator prefix that SHALL be associated with a use case specific data source component stream.

For example:

```
file:///c:/content/example-winxp-xccdf.xml
```

The URL base is: file:///c:/content/

The locator prefix is: example-winxp

The component stream is: xccdf.xml

#### 5.1.2 SCAP Result Data Streams

An SCAP result data stream contains the results of the implementation of one or more of the aforementioned input data streams by an SCAP product. Correlation and aggregation products such as Security Awareness Incident Response tools may consume properly formatted SCAP Result Data Streams to support organizational reporting requirements.

Every SCAP result data stream SHALL use a common locator prefix that is part of a relative URL whose base is the URL of the deployed result file. The notation ‘xxx-’ designates a locator prefix that SHALL

be associated with a use case specific data result component stream. The notation ‘-yyy-‘ designates a locator prefix that MAY be used to differentiate among similar data result component streams.

## 5.2 SCAP Configuration Verification

SCAP enables automated processes to compare system characteristics and settings against an SCAP-expressed checklist. Using such a process, such as that referenced in NIST SP 800-68, *Guide to Securing Microsoft Windows XP Systems for IT Professionals*, a user may confirm compliance and identify deviations from checklists appropriate for relevant operating systems and/or applications.

### 5.2.1 SCAP Configuration Verification Source Data Streams

The following data sources are necessary to support SCAP-compliant configuration verification use cases:

**Table 5-1. SCAP Configuration Verification Data Sources**

Component	Stream Locator	Required/Optional
XCCDF Benchmark	xxxx-xccdf.xml	Required
OVAL Compliance	xxxx-oval.xml	Required
OVAL Patch	xxxx-patches.xml	Optional
OCIL Questionnaire	xxxx-ocil.xml	Optional
CPE Dictionary	xxxx-cpe-dictionary.xml	Required, if applicable
CPE Inventory	xxxx-cpe-oval.xml	Required, if applicable

For an SCAP configuration verification data source to be processed by the appropriate SCAP product:

1. Each Rule specified in the XCCDF benchmark SHALL include an `<ident>` element containing a CCE reference, where an appropriate reference exists.
2. If an `<ident>` is specified in an XCCDF benchmark Rule, then that reference SHALL match the CCE reference found in the associated OVAL Definition(s).
3. If an XCCDF rule references a specific OVAL Definition, then:
  - a. For configuration-related rules:
    - i. The referenced OVAL Definition MUST be a “compliance” class definition.
    - ii. Definitions that are directly or indirectly extended SHALL be limited to “inventory” and “compliance” classes.
    - iii. If an OVAL “compliance” class definition maps to one or more CCE identifiers, the definition SHOULD include `<oval-def:reference>` elements that reference those identifiers using the following format:
 

```
<oval-def:reference source="http://cce.mitre.org"
ref_id="CCE_identifier"/>
```
  - b. For patch-related rules:
    - i. The referenced OVAL Definition MUST be a “patch” class definition.

- ii. Definitions that are directly or indirectly extended SHALL be limited to “inventory” and “patch” classes.
4. An XCCDF benchmark MAY include a “patches up-to-date” rule that references an OVAL patch component stream. If such a rule is used, the OVAL patch component MUST be included in the OVAL patch component as described in Section 4.2.5.3.
5. An XCCDF benchmark MAY enumerate one patch per rule. If this approach is used, a specific OVAL Definition of class “patch” MUST be referenced in the OVAL Patch component stream.
6. An XCCDF rule MAY reference an OCIL questionnaire that MUST be completed with the help of a user.
7. An OVAL Compliance component is an `<oval-def:oval_definitions>` document that specifies definitions for validating the compliance status of target platforms. An OVAL compliance component SHALL specify at least one definition of class “compliance.”

## 5.2.2 SCAP Configuration Verification Results

SCAP enables automated processes to compare system characteristics and settings against an SCAP-expressed checklist. Using such a process, such as that referenced in NIST SP 800-68, *Guide to Securing Microsoft Windows XP Systems for IT Professionals*, a user may confirm compliance and identify deviations from checklists appropriate for relevant operating systems and/or applications.

SCAP-compliant products MUST return Configuration Verification results according to the following conventions:

**Table 5-2. SCAP Configuration Verification Data Results**

Component	Stream Locator	Required/Optional
XCCDF Benchmark	xxxx-yyy-xccdf-res.xml	Required
OVAL Compliance	xxxx-yyy-oval-res.xml	Required
OVAL Patch	xxxx-yyy-patches-res.xml	Required if present in source
OCIL Questionnaire	xxxx-yyy-ocil-res.xml	Required if present in source
CPE Inventory	xxxx-yyy-cpe-oval-res.xml	Required if present in source

SCAP-compliant Configuration Verification results SHALL comply with the following criteria:

1. XCCDF configuration scanning processes SHALL produce XCCDF Results that comply with the XCCDF Results schema.
2. XCCDF Results documents SHALL include a result for each rule that was evaluated during the scan.
3. When producing OVAL and/or OCIL Results, each MUST comply with the Results format per the appropriate specification and:
  - a. OVAL Results documents SHALL include the results that comply with the Oval Results schema of every OVAL Definition used to generate the reported rule results.

- b. OCIL Results documents SHALL include the results that comply with the OCIL schema of every OCIL questionnaire used to evaluate rules.

### 5.3 SCAP Vulnerability Assessment

In the context of SCAP, a vulnerability is defined as a software flaw that introduces a security exposure. SCAP enables interoperability among vulnerability scanners and reporting products to provide consistent detection and reporting of these flaws and supports comprehensive remediation product capabilities.

#### 5.3.1 SCAP Vulnerability Assessment Using XCCDF Source Data Streams

Effective vulnerability assessment using a combination of SCAP components requires the following data sources:

**Table 5-3. SCAP Vulnerability Assessment Data Sources**

Component	Stream Locator	Required/Optional
XCCDF Benchmark	xxxx-xccdf.xml	Required
OVAL Vulnerability	xxxx-oval.xml	Required
OVAL Patch	xxxx-patches.xml	Optional
OCIL Questionnaire	xxxx-ocil.xml	Optional
CPE Dictionary	xxxx-cpe-dictionary.xml	Required, if applicable
CPE Inventory	xxxx-cpe-oval.xml	Required, if applicable

For an SCAP Vulnerability Assessment to be performed by the appropriate SCAP product, the following conditions SHALL be met:

1. XCCDF Vulnerability Scanning SHALL generate an XCCDF Results file. The XCCDF Results document SHALL include a result for each rule that was evaluated during the scan.
2. Each Rule specified in an XCCDF benchmark SHALL include an *<ident>* element containing a CVE reference, where an appropriate reference exists.
3. Each Rule specified in an XCCDF benchmark SHALL reference a specific OVAL vulnerability, patch, or inventory definition, except in cases where no automated mechanism exists to express a check in OVAL.
4. An XCCDF benchmark MAY include a “patches up-to-date” rule that references an OVAL patch component stream. If such a rule is used, the OVAL patch component MUST be included in the OVAL vulnerability component as described in Section 4.2.5.3.
5. An XCCDF benchmark MAY enumerate one patch per rule. If this approach is used, a specific OVAL Definition of class “patch” MUST be referenced in the OVAL Patch component stream.
6. Each Rule specified in an XCCDF benchmark SHALL reference an OCIL questionnaire ONLY for cases where there is no automated mechanism and the checks require human feedback. If a CVE reference is specified in an XCCDF benchmark rule, then that reference SHALL match the CVE reference found in the associated OVAL Definition(s).

### 5.3.2 SCAP Vulnerability Assessment Results Using XCCDF

Effective vulnerability assessment using a combination of SCAP components will return the following results data streams:

**Table 5-4. SCAP Vulnerability Assessment Results**

Component	Stream Locator	Required/Optional
XCCDF Benchmark	xxxx-yyy-xccdf-res.xml	Required
OVAL Vulnerability	xxxx-yyy-oval-res.xml	Required
OVAL Patch	xxxx-yyy-patches-res.xml	Required if present in source
OCIL Questionnaire	xxxx-yyy-ocil-res.xml	Required if present in source
CPE Inventory	xxxx-yyy-cpe-oval-res.xml	Required if present in source

For an SCAP Vulnerability Assessment to be performed by the appropriate SCAP product, the following conditions SHALL be met:

1. When producing OVAL and/or OCIL Results, each MUST comply with the Results format per the appropriate specification and:
  - a. OVAL Results documents SHALL include the results that comply with the Oval Results schema of every OVAL Definition used to generate the reported rule results.
  - b. OCIL Results documents SHALL include the results that comply with the OCIL schema of every OCIL questionnaire used to evaluate rules.

### 5.3.3 SCAP Vulnerability Assessment Using Standalone OVAL Source(s)

For an OVAL-only vulnerability assessment to be processed by the appropriate SCAP product, the following SHALL be present:

1. A Standalone OVAL Vulnerability Data Stream SHALL include an OVAL Vulnerability XML stream component that defines the applied OVAL vulnerability class definitions.
2. OVAL Definitions SHALL include CVE references, if such exist.

### 5.3.4 OVAL Definitions and Vulnerability Assessment

An OVAL vulnerability definition is an `<oval-def:oval_definitions>` document that specifies definitions for assessing the vulnerability status of target platforms. An OVAL vulnerability definition SHALL specify at least one definition of class “vulnerability”. An OVAL vulnerability definition may also reference definitions of class “inventory” or “compliance” that are extended (transitive) by the “vulnerability” class definitions.

If an OVAL “vulnerability” class definition maps to one or more CVE identifiers, the definition SHOULD include `<oval-def:reference>` elements that reference those identifiers using the following format:

```
<oval-def:reference source="CVE" ref_id="CVE_identifier"/>
```

OVAL “vulnerability” class definitions SHOULD also reference source patch identifiers, if they exist.

### 5.3.5 SCAP Vulnerability Assessment Results Using Standalone OVAL

The results of an OVAL-only vulnerability assessment SHALL return results in the following method:

1. OVAL vulnerability data scanning SHALL generate an OVAL Results document that complies with the OVAL Results schema and includes the results of every OVAL Definition contained within the document.
2. The OVAL Results document SHALL include a definition result with supporting system-characteristics data for every definition in the vulnerability data source.

## 5.4 Patch Validation

In the context of SCAP, a patch is defined as a possible remediation for one or more software flaws that introduce a security exposure. SCAP enables interoperability among vulnerability scanners and reporting products to provide consistent detection and reporting of the presence of installed patches and supports comprehensive remediation product capabilities.

### 5.4.1 SCAP Patch Validation Using Standalone OVAL

An OVAL Patch data stream is a singular OVAL component stream that specifies definitions for assessing the patch status of target platforms. The OVAL component stream SHALL specify at least one definition of class “patch”. An OVAL patch definition may also include definitions of class “inventory” that are extended (transitive) by the “patch” class definitions.

If an OVAL “patch” class definition is associated with a source specific identifier (for example, KB numbers for Microsoft patches), these identifiers SHOULD be included in `<oval-def:reference>` elements contained by the definition.

For example:

```
<oval-def:reference source="www.microsoft.com/Patch" ref_id="KB912919"/>
```

If an OVAL “patch” class definition maps to one or more CVE identifiers, the definition MAY include `<oval-def:reference>` elements that reference those identifiers using the following format:

```
<oval-def:reference source="CVE" ref_id="CVE_identifier"/>
```

### 5.4.2 SCAP Patch Validation Results Using Standalone OVAL

The results of an OVAL-only patch assessment SHALL return results in the following method:

1. OVAL patch data scanning SHALL generate an OVAL Results document that complies with the OVAL Results schema and includes the results of every OVAL Definition contained within the document.
2. The OVAL Results document SHALL include a definition result with supporting system-characteristics data for every definition in the OVAL patch component stream.

## 5.5 SCAP Inventory Collection

Organizations require a consistent protocol for integrating inventory information from among a broad range of products, and SCAP provides excellent methods for collecting this data. For example, SCAP

inventory data is an important input to the Risk Management Framework,<sup>39</sup> establishing an effective foundation for system categorization and baseline security controls.

### 5.5.1 SCAP Inventory Collection Data Sources

For SCAP products to collect this inventory information, the following data sources are required:

**Table 5-5. SCAP Inventory Collection**

Component	Stream Locator	Required/Optional
XCCDF Benchmark	xxxx-xccdf.xml	Optional
CPE Dictionary	xxxx-cpe-dictionary.xml	Required, if applicable
CPE Inventory	xxxx-cpe-oval.xml	Required, if applicable

In order for an inventory scan to be processed by the appropriate SCAP product:

1. The inventory data source SHALL include an OVAL Inventory component that defines the applied OVAL inventory class definitions.
2. An OVAL Inventory component SHALL be an `<oval-def:oval_definitions>` document.
3. Specified OVAL Definitions SHALL be “inventory” class definitions for verifying CPE match conditions.

### 5.5.2 SCAP Inventory Collection Data Results

Organizations require a consistent protocol for integrating inventory information from among a broad range of products, and SCAP provides excellent methods for collecting this data. For example, SCAP inventory data is an important input to the Risk Management Framework,<sup>40</sup> establishing an effective foundation for system categorization and baseline security controls. SCAP products should provide the following types of results for inventory collection:

**Table 5-6. SCAP Inventory Collection Results**

Component	Stream Locator	Required/Optional
XCCDF Benchmark	xxxx-yyy-xccdf-res.xml	Optional
CPE Inventory	xxxx-yyy-cpe-oval-res.xml	Required

Inventory results from SCAP products must adhere to the following conventions:

1. OVAL inventory scanning SHALL generate an OVAL Results document that complies with the OVAL Results schema and includes the results of every OVAL Definition used to generate the reported rule results.
2. The results document SHALL include a definition result with supporting system-characteristics data for every definition in the Inventory component.

Within a given `<xccdf:Benchmark>`, `<xccdf:Profile>`, `<xccdf:Group>`, or `<xccdf:Rule>` context, if no platform is defined, the element is considered to apply to the `<xccdf:platform>` of its nearest ancestor. If none is present, the element is considered to apply to all

<sup>39</sup> The Risk Management Framework is explained in NIST SP 800-53 Revision 3 at <http://csrc.nist.gov/publications>.

<sup>40</sup> The Risk Management Framework is explained in NIST SP 800-53 Revision 3 at <http://csrc.nist.gov/publications>.



applicable targets.

## Appendix A—Acronyms and Abbreviations

Selected acronyms and abbreviations used in the guide are defined below.

<b>CCE</b>	Common Configuration Enumeration
<b>CPE</b>	Common Platform Enumeration
<b>CVE</b>	Common Vulnerabilities and Exposures
<b>CVSS</b>	Common Vulnerability Scoring System
<b>DHS</b>	Department of Homeland Security
<b>DISA</b>	Defense Information Systems Agency
<b>DoD</b>	Department of Defense
<b>FDCC</b>	Federal Desktop Core Configuration
<b>FIRST</b>	Forum of Incident Response and Security Teams
<b>FISMA</b>	Federal Information Security Management Act
<b>FSO</b>	DISA Field Security Operations
<b>GPO</b>	Group Policy Object
<b>HIPAA</b>	Health Insurance Portability and Accountability Act
<b>IT</b>	Information Technology
<b>ITL</b>	Information Technology Laboratory
<b>NCSD</b>	National Cyber Security Division
<b>NIST</b>	National Institute of Standards and Technology
<b>NISTIR</b>	National Institute of Standards and Technology Interagency Report
<b>NSA</b>	National Security Agency
<b>NVD</b>	National Vulnerability Database
<b>OCIL</b>	Open Checklist Interactive Language
<b>OMB</b>	Office of Management and Budget
<b>OS</b>	Operating System
<b>OVAL</b>	Open Vulnerability and Assessment Language
<b>PCI</b>	Payment Card Industry
<b>PDI</b>	DISA Potential Discrepancy Item
<b>RFC</b>	Request for Comments
<b>SCAP</b>	Security Content Automation Protocol
<b>SP</b>	Service Pack
<b>SP</b>	Special Publication
<b>STIG</b>	Security Technical Implementation Guide
<b>URI</b>	Uniform Resource Identifier
<b>URL</b>	Uniform Resource Locator
<b>VMS</b>	DISA Vulnerability Management System
<b>XCCDF</b>	eXtensible Configuration Checklist Description Format
<b>XML</b>	eXtensible Markup Language

## Appendix B—References and Other Resources

The lists below provide examples of resources that may be helpful.

- [BAR09] Barrett, M., Johnson, C., Mell, P., Quinn, S., and Scarfone, K., NIST Special Publication 800-117, “Guide to Adopting and Using the Security Content Automation Protocol (SCAP)”, May 2009, <http://csrc.nist.gov/publications/nistpubs/800-117/SP800-117.pdf>
- [BUT09] Buttner, A. and Ziring, N., “Common Platform Enumeration (CPE)—Specification, Version 2.2”, MITRE Corporation, March 11, 2009. [http://cpe.mitre.org/files/cpe-specification\\_2.2.pdf](http://cpe.mitre.org/files/cpe-specification_2.2.pdf)
- [QUI08] Quinn, S. and Ziring, N., NIST Interagency Report 7275 Revision 3, “Specification for the Extensible Configuration Checklist Description Format (XCCDF) Version 1.1.4”, January 2008, <http://csrc.nist.gov/publications/nistir/ir7275r3/NISTIR-7275r3.pdf>

The resources below may be retrieved from the NIST SCAP web site:

- [1] CVE specification and description (<http://scap.nist.gov/revision/1.1/index.html#cve>)
- [2] CCE specification and description (<http://scap.nist.gov/revision/1.1/index.html#cce>)
- [3] CPE specification and description (<http://scap.nist.gov/revision/1.1/index.html#cpe>)
- [4] CVSS specification and description (<http://scap.nist.gov/revision/1.1/index.html#cvss>)
- [5] XCCDF specification and description (<http://scap.nist.gov/revision/1.1/index.html#xccdf>)
- [6] OVAL specification and description (<http://scap.nist.gov/revision/1.1/index.html#oval>)
- [7] OCIL specification and description (<http://scap.nist.gov/revision/1.1/index.html#ocil>)

## Appendix C—SCAP Compliance Verification Data Stream Example

The content in this XML example section has been derived from the NIST SP 800-68 configuration guidance for Windows XP operating systems. This example content is referenced throughout this document as an illustration of how a potential SCAP data stream may be represented. Please note that much of the content from the original source has been removed or changed. The complete, original data stream can be downloaded at <http://web.nvd.nist.gov/view/ncp/repository/checklistDetail?id=76>.

### C.1 XCCDF Benchmark

The following example XCCDF XML instance represents a configuration checklist.

Figure C-1 example-winxp-xccdf.xml

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Benchmark id="Windows-XP-sample" resolved="0" xml:lang="en" style="SCAP_1.1"
3    xmlns="http://checklists.nist.gov/xccdf/1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4    xmlns:cdf="http://checklists.nist.gov/xccdf/1.1" xmlns:cpe="http://cpe.mitre.org/dictionary/2.0"
5    xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:xhtml="http://www.w3.org/1999/xhtml"
6    xmlns:dsig="http://www.w3.org/2000/09/xmldsig#"
7    xsi:schemaLocation="http://checklists.nist.gov/xccdf/1.1 http://nvd.nist.gov/schema/xccdf-
8  1.1.4.xsd
9    http://cpe.mitre.org/dictionary/2.0 http://cpe.mitre.org/files/cpe-dictionary_2.1.xsd">
10   <status date="2009-09-18">draft</status>
11   <title>Windows XP Configuration Verification Example</title>
12   <description>This checklist has been created to assist IT professionals with understanding the
13     requirements and conventions defined in the NIST SP 800-126.</description>
14   <notice id="terms-of-use" xml:lang="en">Do not attempt to implement any of the settings in this
15     guide without first testing them in a non-operational environment. NIST assumes no
16     responsibility whatsoever for its use by other parties, and makes no guarantees, expressed or
17     implied, about its quality, reliability, or any other characteristic. NIST would appreciate
18     acknowledgement if the document and template are used.</notice>
19   <front-matter xml:lang="en">Example front matter</front-matter>
20   <rear-matter xml:lang="en"><xhtml:strong>Trademark
21     Information</xhtml:strong><xhtml:br/><xhtml:br/>Microsoft, Windows, Windows XP, Windows Vista,
22     Internet Explorer, and Windows Firewall are either registered trademarks or trademarks of
23     Microsoft Corporation in the United States and other countries.<xhtml:br/><xhtml:br/>All other
24     names are registered trademarks or trademarks of their respective companies.</rear-matter>
25   <reference href="http://nvd.nist.gov/chklist_detail.cfm?config_id=76">
26     <dc:publisher>National Institute of Standards and Technology</dc:publisher>
27     <dc:identifier>SP 800-68</dc:identifier>
28   </reference>
29   <platform idref="cpe:o:microsoft:windows_xp"/>
30   <version>v0.1</version>
31   <metadata>
32     <dc:creator>National Institute of Standards and Technology</dc:creator>
33     <dc:publisher>National Institute of Standards and Technology</dc:publisher>
34     <dc:contributor>John Doe</dc:contributor>
35   </metadata>
36   <!-- Scoring models supported by this checklist -->
37   <model system="urn:xccdf:scoring:default"/>
38   <model system="urn:xccdf:scoring:flat"/>
39   <Profile id="example-profile-1">
40     <title>Federal Desktop Core Configuration version 1.2.1.0</title>
41     <description>This profile represents guidance outlined in Federal Desktop Core Configuration
42       settings for desktop systems with Windows XP installed.</description>
43     <!-- ..... -->
44     <!-- ' ' ' 2 - FDCC Security Settings ' ' ' -->
45     <!-- ..... -->
46     <!-- Account Lockout Policy Settings -->
47     <select idref="account_lockout_duration" selected="true"/>
48     <!-- Password Policy Settings -->
49     <select idref="minimum_password_length" selected="true"/>
50     <!-- File System Policy -->
51     <select idref="regedit.exePermissions" selected="true"/>

```

```

52 <!-- ..... -->
53 <!-- ' 4 - Fully Patched System ' -->
54 <!-- ..... -->
55 <select idref="security_patches_up_to_date" selected="true"/>
56 <!-- ..... -->
57 <!-- ..... -->
58 <!-- ..... -->
59
60 <refine-value idref="account_lockout_duration_var" selector="900_seconds"/>
61 <refine-value idref="minimum_password_length_var" selector="12_characters"/>
62 </Profile>
63 <!-- This is an example of an XCCDF Group that contains only text. It can be used to represent
64 prose sections of the checklist document to support document generation. -->
65 <Group id="introduction">
66 <title xml:lang="en-US">Introduction</title>
67 <description xml:lang="en-US">This is an example SCAP data stream for the Configuration
68 Verification use case. This data stream is based on the NIST SP 800-68. Please note that much
69 of the content has been removed or changed. The complete, original data stream can be
70 downloaded at:
71 http://web.nvd.nist.gov/view/ncp/repository/checklistDetail?id=76.</description>
72 </Group>
73 <!-- ***** -->
74 <!-- *** 5 - FDCC Security Settings *** -->
75 <!-- ***** -->
76 <Group id="security_settings">
77 <title>Security Settings</title>
78 <description>The following controls must be checked in order to verify compliance.</description>
79 <!-- ~~~~~ -->
80 <!-- ~~~ Account Policies Group ~~~ -->
81 <!-- ~~~~~ -->
82 <Group id="account_policies_group">
83 <title>Account Policies Group</title>
84 <description>todo - description needed</description>
85 <!-- ~~~~~ -->
86 <!-- ~~~~~ Account Lockout Policy Settings ~~~~~ -->
87 <!-- ~~~~~ -->
88 <Group id="account_lockout_policy_settings">
89 <title>Account Lockout Policy Settings</title>
90 <description>Attackers often attempt to gain access to user accounts by guessing passwords.
91 Windows XP can be configured to lock out (disable) an account when too many failed login
92 attempts occur for a single user account in a certain time period. The following account
93 lockout parameters are set in the NIST templates:<xhtml:p/>One of the main challenges in
94 setting account policies is balancing security, functionality, and usability. For example,
95 locking out user accounts after only a few failed logon attempts in a long time period may
96 make it more difficult to gain unauthorized access to accounts by guessing passwords, but
97 may also sharply increase the number of calls to the help desk to unlock accounts
98 accidentally locked by failed attempts from legitimate users. This could also cause more
99 users to write down their passwords or choose easier-to-remember passwords. Organizations
100 should carefully think out such issues before setting Windows XP account
101 policies.</description>
102 <Value id="account_lockout_duration_var" type="number" operator="greater than or equal">
103 <title>Account Lockout Duration</title>
104 <description>The amount of time in seconds that an account is locked before it is
105 automatically unlocked by the system. 15 minutes = 900 seconds A value of 0 means that
106 an administrator must unlock the account.</description>
107 <value>900</value>
108 <value selector="admin_unlock">0</value>
109 <value selector="900_seconds">900</value>
110 <value selector="86400_seconds">86400</value>
111 </Value>
112 <Rule id="account_lockout_duration" selected="false" weight="10.0">
113 <title>Account Lockout Duration</title>
114 <description>The lockout duration specifies how long the user account should be locked out
115 after too many bad logon attempts. This is often set to a low but substantial value
116 (e.g., 15 minutes), for two reasons. First, a legitimate user that is accidentally
117 locked out only has to wait 15 minutes to regain access, instead of asking an
118 administrator to unlock the account. Second, an attacker who is guessing passwords using
119 brute force methods will only be able to try a small number of passwords at a time, then
120 wait 15 minutes before trying any more. This greatly reduces the chances that the brute
121 force attack will be successful.</description>
122 <reference>

```

```

123         <dc:type>GPO</dc:type>
124         <dc:source>Computer Configuration\Windows Settings\Security Settings\Account
125           Policies\Account Lockout Policy</dc:source>
126       </reference>
127       <ident system="http://cve.mitre.org">CVE-2928-0</ident>
128       <ident system="cve.mitre.org/version/4">CVE-980</ident>
129       <check system="http://oval.mitre.org/XMLSchema/oval-definitions-5">
130         <check-export value-id="account_lockout_duration_var"
131           export-name="oval:gov.nist.fdcc.xp:var:15"/>
132         <check-content-ref href="example-winxp-oval.xml" name="oval:gov.nist.fdcc.xp:def:23"/>
133       </check>
134     </Rule>
135   </Group>
136   <!-- ~~~~~~ Password Policy Settings ~~~~~~ -->
137   <!-- ~~~~~~ Password Policy Settings ~~~~~~ -->
138   <!-- ~~~~~~ Password Policy Settings ~~~~~~ -->
139   <Group id="password_policy_settings">
140     <title>Password Policies</title>
141     <description>In addition to educating users regarding the selection and use of good
142       passwords, it is also important to set password parameters so that passwords are
143       sufficiently strong. This reduces the likelihood of an attacker guessing or cracking
144       passwords to gain unauthorized access to the system. As described in Section 3.2.1, NIST
145       recommends the use of NTLM v2 or Kerberos instead of LM or NTLM v1 for authentication.
146       Windows XP offers the same password parameters as Windows 2000. The following parameters
147       are specified in the NIST templates:</description>
148     <Value id="minimum_password_length_var" type="number" operator="greater than or equal">
149       <title>Minimum Password Length</title>
150       <description>The minimum number of characters required for a password</description>
151       <value>8</value>
152       <value selector="8_characters">8</value>
153       <value selector="9_characters">9</value>
154       <value selector="12_characters">12</value>
155     </Value>
156     <Rule id="minimum_password_length" selected="false" weight="10.0">
157       <title>Minimum Password Length</title>
158       <description>This setting specifies the minimum length of a password in characters. The
159       rationale behind this setting is that longer passwords are more difficult to guess and
160       crack than shorter passwords. The downside is that longer passwords are often more
161       difficult for users to remember. Organizations that want to set a relatively large
162       minimum password length should encourage their users to use passphrases, which may be
163       easier to remember than conventional passwords.</description>
164       <reference>
165         <dc:type>GPO</dc:type>
166         <dc:source>Computer Configuration\Windows Settings\Security Settings\Account
167           Policies\Password Policy</dc:source>
168       </reference>
169       <ident system="http://cve.mitre.org">CVE-2981-9</ident>
170       <ident system="cve.mitre.org/version/4">CVE-100</ident>
171       <check system="http://oval.mitre.org/XMLSchema/oval-definitions-5">
172         <check-export value-id="minimum_password_length_var"
173           export-name="oval:gov.nist.fdcc.xp:var:12"/>
174         <check-content-ref href="example-winxp-oval.xml" name="oval:gov.nist.fdcc.xp:def:19"/>
175       </check>
176     </Rule>
177   </Group>
178 </Group>
179 <!-- ~~~~~~ File Permissions Group ~~~~~~ -->
180 <!-- ~~~~~~ File Permissions Group ~~~~~~ -->
181 <!-- ~~~~~~ File Permissions Group ~~~~~~ -->
182 <Group id="file_permissions_group">
183   <title>File Permission Settings</title>
184   <description>This group checks the permissions of specified files.</description>
185   <Rule id="regedit.exePermissions" selected="false" weight="10.0">
186     <title>regedit.exe Permissions</title>
187     <description>Failure to properly configure ACL file and directory permissions, allows the
188       possibility of unauthorized and anonymous modification to the operating system and
189       installed applications.</description>
190     <reference>
191       <dc:type>GPO</dc:type>
192       <dc:source>Computer Configuration\Windows Settings\Security Settings\File
193         System</dc:source>

```

```

194     </reference>
195     <ident system="http://cce.mitre.org">CCE-2175-8</ident>
196     <ident system="cce.mitre.org/version/4">CCE-795</ident>
197     <check system="http://oval.mitre.org/XMLSchema/oval-definitions-5">
198       <check-content-ref href="example-winxp-oval.xml" name="oval:gov.nist.fdcc.xp:def:146"/>
199     </check>
200   </Rule>
201 </Group>
202 </Group>
203 <!-- ~~~~~~ -->
204 <!-- ~~~ Local Policies Group ~~~ -->
205 <!-- ~~~~~~ -->
206 <Group id="local_policies_group">
207   <title>Local Policies Group</title>
208   <description>...</description>
209   <!-- ~~~~~~ -->
210   <!-- Security Options Settings -->
211   <!-- ~~~~~~ -->
212   <Group id="security_options_settings">
213     <title>Security Options Settings</title>
214     <description>..</description>
215     <Rule id="anonymous_sid_name_translation" selected="false" weight="10.0" role="unchecked">
216       <title>Network access: Allow anonymous SID-Name translation</title>
217       <description>Determines if an anonymous user can request security identifier
218         (SID) attributes for another user or use a SID to get the corresponding
219         username.
220     </description>
221     <reference>
222       <dc:type>GPO</dc:type>
223       <dc:source>Computer Configuration\Windows Settings\Security Settings\Local
224         Policies\Security Options</dc:source>
225     </reference>
226     <ident system="http://cce.mitre.org">CCE-2973-6</ident>
227     <ident system="cce.mitre.org/version/4">CCE-953</ident>
228     <check system="http://www.mitre.org/ocil/2">
229       <check-content-ref href="example-winxp-ocil.xml"
230         name="ocil:mitre.org:questionnaire:1"/>
231     </check>
232   </Rule>
233 </Group>
234 </Group>
235 <!-- ***** -->
236 <!-- *** 7 - Security Patches *** -->
237 <!-- ***** -->
238 <Group id="security_patches">
239   <title>Security Patches</title>
240   <description>Securing a given computer has become increasingly important. As such, it is
241     essential to keep a host up to current patch levels to eliminate known vulnerabilities and
242     weaknesses. In conjunction with antivirus software and a personal firewall, patching goes a
243     long way to securing a host against outside attacks and exploitation. Microsoft provides two
244     mechanisms for distributing security updates: Automatic Updates and Microsoft Update. In
245     smaller environments, either method may be sufficient for keeping systems current with
246     patches. Other environments typically have a software change management control process or a
247     patch management program that tests patches before deploying them; distribution may then occur
248     through local Windows Update Services (WUS) or Windows Server Update Services (WSUS) servers,
249     which provide approved security patches for use by the Automatic Updates
250     feature.</description>
251   <Rule id="security_patches_up_to_date" selected="false" weight="10.0">
252     <title>Security Patches Up-To-Date</title>
253     <description>Keep systems up to current patch levels to eliminate known vulnerabilities and
254       weaknesses.</description>
255     <check system="http://oval.mitre.org/XMLSchema/oval-definitions-5">
256       <check-content-ref href="http://nvd.nist.gov/scap/content/fdcc-winxp-patches.xml"/>
257       <check-content-ref href="example-winxp-patches.xml"/>
258     </check>
259   </Rule>
260 </Group>
261 <!-- ===== -->
262 <!-- ===== -->
263 <!-- ===== -->
264 </Benchmark>

```

## C.2 OVAL Compliance

The following OVAL XML instance represents compliance definitions used to evaluate the XCCDF checklist from the previous section.

Figure C-2 example-winxp-oval.xml

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <oval_definitions xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5"
3    xmlns:oval="http://oval.mitre.org/XMLSchema/oval-common-5"
4    xmlns:oval-def="http://oval.mitre.org/XMLSchema/oval-definitions-5"
5    xmlns:win-def="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
6    xmlns:ind-def="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
7    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
8    xsi:schemaLocation="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows
9    http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/windows-
10    definitions-schema.xsd
11    http://oval.mitre.org/XMLSchema/oval-definitions-5#independent
12    http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/independent-
13    definitions-schema.xsd
14    http://oval.mitre.org/XMLSchema/oval-definitions-5
15    http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/oval-
16    definitions-schema.xsd
17    http://oval.mitre.org/XMLSchema/oval-common-5
18    http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/oval-common-
19    schema.xsd">
20    <generator>
21      <oval:product_name>National Institute of Standards and Technology</oval:product_name>
22      <oval:schema_version>5.6</oval:schema_version>
23      <oval:timestamp>2009-04-08T15:04:22.000-05:00</oval:timestamp>
24    </generator>
25    <!-- ===== -->
26    <!-- DEFINITIONS ===== -->
27    <!-- ===== -->
28    <definitions>
29      <definition id="oval:gov.nist.fdcc.xp:def:23" version="1" class="compliance">
30        <metadata>
31          <title>Account Lockout Duration</title>
32          <affected family="windows">
33            <platform>Microsoft Windows XP</platform>
34          </affected>
35          <reference source="http://cve.mitre.org" ref_id="CVE-2009-3555"/>
36          <reference source="cve.mitre.org/version/4" ref_id="CVE-980"/>
37          <description>This definition verifies that locked accounts remains locked for the defined
38            number of minutes before they are automatically unlocked.</description>
39        </metadata>
40        <criteria>
41          <extend_definition comment="Microsoft Windows XP is installed"
42            definition_ref="oval:gov.nist.fdcc.xp:def:2"/>
43          <criteria operator="OR">
44            <criterion
45              comment="Account Lockout Duration is set to keep accounts locked for at least the
46              defined number of minutes"
47              test_ref="oval:gov.nist.fdcc.xp:tst:19"/>
48            <criterion
49              comment="Account Lockout Duration is set to keep accounts locked until an
50              administrator unlocks them"
51              test_ref="oval:gov.nist.fdcc.xp:tst:1911"/>
52          </criteria>
53        </criteria>
54      </definition>
55      <definition id="oval:gov.nist.fdcc.xp:def:19" version="1" class="compliance">
56        <metadata>
57          <title>Minimum Password Length</title>
58          <affected family="windows">
59            <platform>Microsoft Windows XP</platform>
60          </affected>
61          <reference source="http://cve.mitre.org" ref_id="CVE-2009-3555"/>

```



```

62     <reference source="cce.mitre.org/version/4" ref_id="CCE-100"/>
63     <description>Minimum password length is the profile defined number of
64         characters</description>
65 </metadata>
66 <criteria>
67     <extend_definition comment="Microsoft Windows XP is installed"
68         definition_ref="oval:gov.nist.fdcc.xp:def:2"/>
69     <criteria comment="Minimum password length is profile defined"
70         test_ref="oval:gov.nist.fdcc.xp:tst:15"/>
71 </criteria>
72 </definition>
73 <definition id="oval:gov.nist.fdcc.xp:def:146" version="1" class="compliance">
74     <metadata>
75         <title>Administrators and System User Have Full Access to the SYSTEMROOT/regedit.exe
76             File</title>
77         <affected family="windows">
78             <platform>Microsoft Windows XP</platform>
79         </affected>
80         <reference source="http://cce.mitre.org" ref_id="CCE-2175-8"/>
81         <reference source="cce.mitre.org/version/4" ref_id="CCE-795"/>
82         <description>The Administrators group and the System user should have full access to the
83             SYSTEMROOT/regedit.exe file and all other users should have no file access
84             privileges</description>
85     </metadata>
86     <criteria>
87         <extend_definition comment="Microsoft Windows XP is installed"
88             definition_ref="oval:gov.nist.fdcc.xp:def:2"/>
89         <criteria operator="AND">
90             <criteria
91                 comment="The Administrators group is granted full access to the file regedit.exe"
92                 test_ref="oval:gov.nist.fdcc.xp:tst:248"/>
93             <criteria comment="The System user is granted full access to the file regedit.exe"
94                 test_ref="oval:gov.nist.fdcc.xp:tst:249"/>
95             <criteria
96                 comment="There are no access privileges to file regedit.exe by users not part of the
97 Administrators group or the System user"
98                 test_ref="oval:gov.nist.fdcc.xp:tst:250"/>
99             </criteria>
100         </criteria>
101     </definition>
102
103 <!--=====
104 <!--===== EXTENDED DEFINITIONS =====
105 <!--=====
106 <definition id="oval:gov.nist.fdcc.xp:def:2" version="1" class="inventory">
107     <metadata>
108         <title>Microsoft Windows XP is installed</title>
109         <affected family="windows">
110             <platform>Microsoft Windows XP</platform>
111         </affected>
112         <description>Microsoft Windows XP is installed</description>
113     </metadata>
114     <criteria>
115         <criteria comment="the installed operating system is part of the Microsoft Windows
116 family"
117             test_ref="oval:gov.nist.fdcc.xp:tst:6"/>
118         <criteria comment="Microsoft Windows XP is installed"
119             test_ref="oval:gov.nist.fdcc.xp:tst:7"/>
120     </criteria>
121 </definition>
122 </definitions>
123 <!-- =====
124 <!-- ===== TESTS =====
125 <!-- =====
126 <tests>
127     <family_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
128         id="oval:gov.nist.fdcc.xp:tst:6" version="1"
129         comment="the installed operating system is part of the Microsoft Windows family"
130         check_existence="at_least_one_exists" check="only one">
131         <object object_ref="oval:gov.nist.fdcc.xp:obj:3"/>
132         <state state_ref="oval:gov.nist.fdcc.xp:ste:14"/>

```

```

133 </family_test>
134 <registry_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
135   id="oval:gov.nist.fdcc.xp:tst:7" version="1" comment="Microsoft Windows XP is installed"
136   check_existence="at_least_one_exists" check="at least one">
137   <object object_ref="oval:gov.nist.fdcc.xp:obj:4"/>
138   <state state_ref="oval:gov.nist.fdcc.xp:ste:15"/>
139 </registry_test>
140 <passwordpolicy_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
141   id="oval:gov.nist.fdcc.xp:tst:15" version="1"
142   comment="Minimum password length is profile defined" check_existence="at_least_one_exists"
143   check="all">
144   <object object_ref="oval:gov.nist.fdcc.xp:obj:8"/>
145   <state state_ref="oval:gov.nist.fdcc.xp:ste:20"/>
146 </passwordpolicy_test>
147 <lockoutpolicy_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
148   id="oval:gov.nist.fdcc.xp:tst:19" version="1"
149   comment="Account Lockout Duration is set to keep accounts locked for at least the defined
150 number of minutes"
151   check_existence="at_least_one_exists" check="all">
152   <object object_ref="oval:gov.nist.fdcc.xp:obj:9"/>
153   <state state_ref="oval:gov.nist.fdcc.xp:ste:25"/>
154 </lockoutpolicy_test>
155 <lockoutpolicy_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
156   id="oval:gov.nist.fdcc.xp:tst:1911" version="1"
157   comment="Account Lockout Duration is set to keep accounts locked until an administrator
158 unlocks them"
159   check_existence="at_least_one_exists" check="all">
160   <object object_ref="oval:gov.nist.fdcc.xp:obj:9"/>
161   <state state_ref="oval:gov.nist.fdcc.xp:ste:2511"/>
162 </lockoutpolicy_test>
163 <fileeffectiverights53_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
164 5#windows"
165   id="oval:gov.nist.fdcc.xp:tst:248" version="1"
166   comment="The Administrators group is granted full access to the file regedit.exe"
167   check_existence="any_exist" check="all">
168   <object object_ref="oval:gov.nist.fdcc.xp:obj:155"/>
169   <state state_ref="oval:gov.nist.fdcc.xp:ste:51"/>
170 </fileeffectiverights53_test>
171 <fileeffectiverights53_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
172 5#windows"
173   id="oval:gov.nist.fdcc.xp:tst:249" version="1"
174   comment="The System user is granted full access to the file regedit.exe"
175   check_existence="any_exist" check="all">
176   <object object_ref="oval:gov.nist.fdcc.xp:obj:156"/>
177   <state state_ref="oval:gov.nist.fdcc.xp:ste:51"/>
178 </fileeffectiverights53_test>
179 <fileeffectiverights53_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
180 5#windows"
181   id="oval:gov.nist.fdcc.xp:tst:250" version="1"
182   comment="There are no access privileges to file regedit.exe by users not part of the
183 Administrators group or the System user"
184   check_existence="any_exist" check="all">
185   <object object_ref="oval:gov.nist.fdcc.xp:obj:157"/>
186   <state state_ref="oval:gov.nist.fdcc.xp:ste:52"/>
187 </fileeffectiverights53_test>
188 </tests>
189 <!-- ===== -->
190 <!-- ===== OBJECTS ===== -->
191 <!-- ===== -->
192 <objects>
193   <family_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
194     id="oval:gov.nist.fdcc.xp:obj:3" version="1"/>
195   <registry_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
196     id="oval:gov.nist.fdcc.xp:obj:4" version="1">
197     <hive>HKEY_LOCAL_MACHINE</hive>
198     <key>SOFTWARE\Microsoft\Windows NT\CurrentVersion</key>
199     <name>CurrentVersion</name>
200   </registry_object>
201   <passwordpolicy_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
202     id="oval:gov.nist.fdcc.xp:obj:8" version="1"/>
203   <lockoutpolicy_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"

```

```

204     id="oval:gov.nist.fdcc.xp:obj:9" version="1"/>
205 <registry_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
206   id="oval:gov.nist.fdcc.xp:obj:79" version="1">
207   <hive>HKEY_LOCAL_MACHINE</hive>
208   <key>SOFTWARE\Microsoft\Windows NT\CurrentVersion</key>
209   <name>SystemRoot</name>
210 </registry_object>
211 <fileeffectiverights53_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
212 5#windows"
213   id="oval:gov.nist.fdcc.xp:obj:155" version="1">
214   <path datatype="string" var_ref="oval:gov.nist.fdcc.xp:var:4"/>
215   <filename>regedit.exe</filename>
216   <trustee_sid>S-1-5-32-544</trustee_sid>
217 </fileeffectiverights53_object>
218 <fileeffectiverights53_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
219 5#windows"
220   id="oval:gov.nist.fdcc.xp:obj:156" version="1">
221   <path datatype="string" var_ref="oval:gov.nist.fdcc.xp:var:4"/>
222   <filename>regedit.exe</filename>
223   <trustee_sid>S-1-5-18</trustee_sid>
224 </fileeffectiverights53_object>
225 <fileeffectiverights53_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
226 5#windows"
227   id="oval:gov.nist.fdcc.xp:obj:157" version="1">
228   <set xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5"
229   set_operator="INTERSECTION">
230     <set set_operator="COMPLEMENT">
231       <object_reference>oval:gov.nist.fdcc.xp:obj:318</object_reference>
232       <object_reference>oval:gov.nist.fdcc.xp:obj:156</object_reference>
233     </set>
234     <set set_operator="COMPLEMENT">
235       <object_reference>oval:gov.nist.fdcc.xp:obj:318</object_reference>
236       <object_reference>oval:gov.nist.fdcc.xp:obj:158</object_reference>
237     </set>
238   </set>
239 </fileeffectiverights53_object>
240 <fileeffectiverights53_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
241 5#windows"
242   id="oval:gov.nist.fdcc.xp:obj:318" version="1">
243   <behaviors resolve_group="true"/>
244   <path datatype="string" var_ref="oval:gov.nist.fdcc.xp:var:4"/>
245   <filename>regedit.exe</filename>
246   <trustee_sid operation="pattern match">.*</trustee_sid>
247 </fileeffectiverights53_object>
248 <fileeffectiverights53_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
249 5#windows"
250   id="oval:gov.nist.fdcc.xp:obj:158" version="1">
251   <behaviors resolve_group="true"/>
252   <path datatype="string" var_ref="oval:gov.nist.fdcc.xp:var:4"/>
253   <filename>regedit.exe</filename>
254   <trustee_sid>S-1-5-32-544</trustee_sid>
255 </fileeffectiverights53_object>
256 </objects>
257 <!-- =====>
258 <!-- ===== STATES =====>
259 <!-- =====>
260 <states>
261   <family_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
262     id="oval:gov.nist.fdcc.xp:ste:14" version="1">
263     <family>windows</family>
264   </family_state>
265   <passwordpolicy_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
266     id="oval:gov.nist.fdcc.xp:ste:20" version="1">
267     <min_passwd_len datatype="int" operation="greater than or equal"
268       var_ref="oval:gov.nist.fdcc.xp:var:12"/>
269   </passwordpolicy_state>
270   <lockoutpolicy_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
271     id="oval:gov.nist.fdcc.xp:ste:25" version="1">
272     <lockout_duration datatype="int" operation="greater than or equal"
273       var_ref="oval:gov.nist.fdcc.xp:var:15"/>
274   </lockoutpolicy_state>

```

```

275 <lockoutpolicy_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
276   id="oval:gov.nist.fdcc.xp:ste:2511" version="1">
277   <lockout_duration datatype="int">-1</lockout_duration>
278 </lockoutpolicy_state>
279 <registry_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
280   id="oval:gov.nist.fdcc.xp:ste:15" version="1">
281   <value datatype="string">5.1</value>
282 </registry_state>
283 <fileeffectiverights53_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
284 5#windows"
285   id="oval:gov.nist.fdcc.xp:ste:51" version="1"
286   comment="specified account is granted full control">
287   <standard_delete datatype="boolean">1</standard_delete>
288   <standard_read_control datatype="boolean">1</standard_read_control>
289   <standard_write_dac datatype="boolean">1</standard_write_dac>
290   <standard_write_owner datatype="boolean">1</standard_write_owner>
291   <standard_synchronize datatype="boolean">1</standard_synchronize>
292   <file_read_data datatype="boolean">1</file_read_data>
293   <file_write_data datatype="boolean">1</file_write_data>
294   <file_append_data datatype="boolean">1</file_append_data>
295   <file_read_ea datatype="boolean">1</file_read_ea>
296   <file_write_ea datatype="boolean">1</file_write_ea>
297   <file_execute datatype="boolean">1</file_execute>
298   <file_delete_child datatype="boolean">1</file_delete_child>
299   <file_read_attributes datatype="boolean">1</file_read_attributes>
300   <file_write_attributes datatype="boolean">1</file_write_attributes>
301 </fileeffectiverights53_state>
302 <fileeffectiverights53_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-
303 5#windows"
304   id="oval:gov.nist.fdcc.xp:ste:52" version="1"
305   comment="specified account has no access privileges">
306   <standard_delete datatype="boolean">0</standard_delete>
307   <standard_read_control datatype="boolean">0</standard_read_control>
308   <standard_write_dac datatype="boolean">0</standard_write_dac>
309   <standard_write_owner datatype="boolean">0</standard_write_owner>
310   <standard_synchronize datatype="boolean">0</standard_synchronize>
311   <access_system_security datatype="boolean">0</access_system_security>
312   <generic_read datatype="boolean">0</generic_read>
313   <generic_write datatype="boolean">0</generic_write>
314   <generic_execute datatype="boolean">0</generic_execute>
315   <generic_all datatype="boolean">0</generic_all>
316   <file_read_data datatype="boolean">0</file_read_data>
317   <file_write_data datatype="boolean">0</file_write_data>
318   <file_append_data datatype="boolean">0</file_append_data>
319   <file_read_ea datatype="boolean">0</file_read_ea>
320   <file_write_ea datatype="boolean">0</file_write_ea>
321   <file_execute datatype="boolean">0</file_execute>
322   <file_delete_child datatype="boolean">0</file_delete_child>
323   <file_read_attributes datatype="boolean">0</file_read_attributes>
324   <file_write_attributes datatype="boolean">0</file_write_attributes>
325 </fileeffectiverights53_state>
326 </states>
327 <!-- ===== -->
328 <!-- ===== VARIABLES ===== -->
329 <!-- ===== -->
330 <variables>
331   <local_variable id="oval:gov.nist.fdcc.xp:var:1" version="1"
332     comment="Windows system32 directory" datatype="string">
333     <concat>
334       <object_component object_ref="oval:gov.nist.fdcc.xp:obj:79" item_field="value"/>
335       <literal_component>\system32</literal_component>
336     </concat>
337   </local_variable>
338   <local_variable id="oval:gov.nist.fdcc.xp:var:4" version="1" comment="Windows directory"
339     datatype="string">
340     <object_component object_ref="oval:gov.nist.fdcc.xp:obj:79" item_field="value"/>
341   </local_variable>
342   <external_variable id="oval:gov.nist.fdcc.xp:var:12" version="1"
343     comment="minimum password length" datatype="int"/>
344   <external_variable id="oval:gov.nist.fdcc.xp:var:15" version="1"
345     comment="Account lockout duration" datatype="int"/>

```

```

346 </variables>
347 <!-- ===== -->
348 <!-- ===== -->
349 <!-- ===== -->
350 </oval_definitions>

```

### C.3 OVAL Patch

The following OVAL XML instance represents patch definitions used to assess the `<xccdf:Rule>` element from section C.1 lines 223-226.

Figure C-3 example-winxp-patches.xml

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <oval_definitions xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5"
3    xmlns:oval="http://oval.mitre.org/XMLSchema/oval-common-5"
4    xmlns:oval-def="http://oval.mitre.org/XMLSchema/oval-definitions-5"
5    xmlns:ind-def="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
6    xmlns:win-def="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
7    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
8    xsi:schemaLocation="http://oval.mitre.org/XMLSchema/oval-common-5
9    http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/oval-common-
10    schema.xsd
11    http://oval.mitre.org/XMLSchema/oval-definitions-5
12    http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/oval-
13    definitions-schema.xsd
14    http://oval.mitre.org/XMLSchema/oval-definitions-5#windows
15    http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/windows-
16    definitions-schema.xsd
17    http://oval.mitre.org/XMLSchema/oval-definitions-5#independent
18    http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/independent-
19    definitions-schema.xsd">
20    <generator>
21      <oval:product_name>National Institute of Standards and Technology</oval:product_name>
22      <oval:schema_version>5.3</oval:schema_version>
23      <oval:timestamp>2009-08-21T08:34:35.000-05:00</oval:timestamp>
24    </generator>
25    <!-- ===== -->
26    <!-- ===== DEFINITIONS ===== -->
27    <!-- ===== -->
28    <definitions>
29      <definition id="oval:gov.nist.fdcc.patch:def:5" version="0" class="patch">
30        <metadata>
31          <title>MS05-013: Vulnerability in the DHTML Editing Component ActiveX Control Could Allow
32            Remote Code Execution (891781)</title>
33          <affected family="windows">
34            <platform>Microsoft Windows XP</platform>
35            <product>Microsoft Internet Explorer</product>
36          </affected>
37          <reference source="Microsoft" ref_id="MS05-013"
38            ref_url="http://www.microsoft.com/technet/security/bulletin/ms05-013.mspx"/>
39          <reference source="Microsoft" ref_id="KB891781"
40            ref_url="http://support.microsoft.com/kb/891781"/>
41          <reference source="Bugtraq ID" ref_id="11950"
42            ref_url="http://www.securityfocus.com/bid/11950"/>
43          <reference source="CERT-VN" ref_id="VU#356600"
44            ref_url="http://www.kb.cert.org/vuls/id/356600"/>
45          <reference source="CIAC" ref_id="p-126"
46            ref_url="http://www.ciac.org/ciac/bulletins/p-126.shtml"/>
47          <reference source="CVE" ref_id="CVE-2004-1319"
48            ref_url="http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2004-1319"/>
49          <reference source="OVAL" ref_id="oval:org.mitre.oval:def:3851"
50            ref_url="http://oval.mitre.org/repository/data/getDef?id=oval:org.mitre.oval:def:3851"/>
51          <reference source="OVAL" ref_id="oval:org.mitre.oval:def:1701"
52            ref_url="http://oval.mitre.org/repository/data/getDef?id=oval:org.mitre.oval:def:1701"/>
53
54          <reference source="OVAL" ref_id="oval:org.mitre.oval:def:1701"
55            ref_url="http://oval.mitre.org/repository/data/getDef?id=oval:org.mitre.oval:def:1701"/>

```

```

55     <reference source="OVAL" ref_id="oval:org.mitre.oval:def:4758"
56
57 ref_url="http://oval.mitre.org/repository/data/getDef?id=oval:org.mitre.oval:def:4758"/>
58     <reference source="OVAL" ref_id="oval:org.mitre.oval:def:1114"
59
60 ref_url="http://oval.mitre.org/repository/data/getDef?id=oval:org.mitre.oval:def:1114"/>
61     <reference source="OVAL" ref_id="oval:org.mitre.oval:def:3464"
62
63 ref_url="http://oval.mitre.org/repository/data/getDef?id=oval:org.mitre.oval:def:3464"/>
64     <description>Microsoft has released MS05-013 to address security issues in Microsoft
65 Internet Explorer as documented by CVE-2004-1319.</description>
66 </metadata>
67 <criteria comment="Software section" operator="AND">
68     <extend_definition comment="Microsoft Windows XP (32-bit) SP2 is installed"
69     definition_ref="oval:gov.nist.fdcc.patch:def:115276"/>
70     <criteria comment="the version of dhtmlmed.ocx is less than 6.1.0.9232" negate="false"
71     test_ref="oval:org.mitre.oval:tst:427"/>
72     <criteria comment="the patch kb891781 is installed (Hotfix key)" negate="true"
73     test_ref="oval:org.mitre.oval:tst:1151"/>
74 </criteria>
75 </definition>
76 <definition id="oval:gov.nist.fdcc.patch:def:1784" version="2" class="patch">
77 <metadata>
78     <title>MS07-050: Vulnerability in Vector Markup Language Could Allow Remote Code
79 Execution
80     (938127)</title>
81     <affected family="windows">
82     <platform>Microsoft Windows XP</platform>
83     <platform>Microsoft Windows Vista</platform>
84     <product>Microsoft Internet Explorer</product>
85 </affected>
86     <reference source="Microsoft" ref_id="MS07-050"
87     ref_url="http://www.microsoft.com/technet/security/bulletin/MS07-050.mspx"/>
88     <reference source="Microsoft" ref_id="938127"
89     ref_url="http://support.microsoft.com/kb/938127"/>
90     <reference source="CVE" ref_id="CVE-2007-1749"
91     ref_url="http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-1749"/>
92     <reference source="OVAL" ref_id="oval:org.mitre.oval:def:1784"
93
94 ref_url="http://oval.mitre.org/repository/data/getDef?id=oval:org.mitre.oval:def:1784"/>
95     <description>Microsoft has released MS07-050 to address security issues in the Vector
96 Markup
97 Language (VML) implementation in Windows as documented by CVE-2007-1749.</description>
98 </metadata>
99 <criteria operator="OR">
100     <criteria comment="IE 6 on Win XP SP2" operator="AND">
101     <extend_definition comment="Microsoft Windows XP (32-bit) SP2 is installed"
102     definition_ref="oval:gov.nist.fdcc.patch:def:115276"/>
103     <extend_definition definition_ref="oval:org.mitre.oval:def:563"
104     comment="Internet Explorer 6 is installed"/>
105     <criteria comment="the version of vgx.dll is less than 6.0.2900.3164"
106     test_ref="oval:org.mitre.oval:tst:3856"/>
107 </criteria>
108     <criteria comment="IE 7 on Win XP SP2" operator="AND">
109     <extend_definition comment="Microsoft Windows XP (32-bit) SP2 is installed"
110     definition_ref="oval:gov.nist.fdcc.patch:def:115276"/>
111     <extend_definition definition_ref="oval:org.mitre.oval:def:627"
112     comment="Internet Explorer 7 is installed"/>
113     <criteria comment="the version of vgx.dll is less than 7.0.6000.20628"
114     test_ref="oval:org.mitre.oval:tst:4182"/>
115 </criteria>
116     <criteria comment="IE 6 on Win XP SP2 (64-bit)" operator="AND">
117     <extend_definition comment="Microsoft Windows XP SP2 (64-bit) is installed"
118     definition_ref="oval:gov.nist.fdcc.patch:def:115277"/>
119     <extend_definition comment="Internet Explorer 6 is installed"
120     definition_ref="oval:org.mitre.oval:def:563"/>
121     <criteria comment="the version of vgx.dll is less than 6.0.3790.4106"
122     test_ref="oval:org.mitre.oval:tst:3422"/>
123 </criteria>
124     <criteria comment="IE 7 on Win XP SP2 (64-bit)" operator="AND">
125     <extend_definition comment="Microsoft Windows XP SP2 (64-bit) is installed"

```



```

126         definition_ref="oval:gov.nist.fdcc.patch:def:115277"/>
127     <extend_definition definition_ref="oval:org.mitre.oval:def:627"
128         comment="Internet Explorer 7 is installed"/>
129     <criteria comment="the version of vgx.dll is less than 7.0.6000.20628"
130         test_ref="oval:org.mitre.oval:tst:4182"/>
131     </criteria>
132 </criteria>
133 </definition>
134
135 <!-- ===== -->
136 <!-- ===== EXTENDED DEFINITIONS ===== -->
137 <!-- ===== -->
138 <definition id="oval:gov.nist.fdcc.patch:def:115275" version="0" class="inventory">
139     <metadata>
140         <title>Microsoft Windows XP is installed</title>
141         <affected family="windows">
142             <platform>Microsoft Windows XP</platform>
143         </affected>
144         <reference source="CPE" ref_id="cpe:/o:microsoft:windows_xp"/>
145         <description>The operating system installed on the system is Microsoft Windows
146             XP.</description>
147     </metadata>
148     <criteria operator="AND">
149         <criteria comment="the installed operating system is part of the Microsoft Windows
150 family"
151             test_ref="oval:org.mitre.oval:tst:99"/>
152         <criteria comment="Windows XP is installed"
153             test_ref="oval:gov.nist.fdcc.patch:tst:57914"/>
154         <criteria comment="Current version = 5.1 or 5.2"
155             test_ref="oval:gov.nist.fdcc.patch:tst:115300"/>
156     </criteria>
157 </definition>
158 <definition id="oval:gov.nist.fdcc.patch:def:115276" version="0" class="inventory">
159     <metadata>
160         <title>Microsoft Windows XP (32-bit) SP2 is installed</title>
161         <affected family="windows">
162             <platform>Microsoft Windows XP</platform>
163         </affected>
164         <reference source="CPE" ref_id="cpe:/o:microsoft:windows_xp::sp2:x86"/>
165         <description>A version of Microsoft Windows XP (32-bit) SP2 is installed.</description>
166     </metadata>
167     <criteria operator="AND">
168         <extend_definition comment="Microsoft Windows XP is installed"
169             definition_ref="oval:gov.nist.fdcc.patch:def:115275"/>
170         <criteria comment="a version of Windows for the x86 architecture is installed"
171             test_ref="oval:org.mitre.oval:tst:3823"/>
172         <criteria comment="Win2K/XP/2003 service pack 2 is installed"
173             test_ref="oval:org.mitre.oval:tst:3019"/>
174     </criteria>
175 </definition>
176 <definition id="oval:gov.nist.fdcc.patch:def:115277" version="0" class="inventory">
177     <metadata>
178         <title>Microsoft Windows XP (64-bit) SP2 is installed</title>
179         <affected family="windows">
180             <platform>Microsoft Windows XP</platform>
181         </affected>
182         <reference source="CPE" ref_id="cpe:/o:microsoft:windows_xp::sp2:x64"/>
183         <description>A version of Microsoft Windows XP (64-bit) SP2 is installed.</description>
184     </metadata>
185     <criteria operator="AND">
186         <extend_definition comment="Microsoft Windows XP is installed"
187             definition_ref="oval:gov.nist.fdcc.patch:def:115275"/>
188         <criteria comment="a version of Windows for the x64 architecture is installed"
189             test_ref="oval:org.mitre.oval:tst:3653"/>
190         <criteria comment="Win2K/XP/2003 service pack 2 is installed"
191             test_ref="oval:org.mitre.oval:tst:3019"/>
192     </criteria>
193 </definition>
194 <definition id="oval:org.mitre.oval:def:563" version="3" class="inventory">
195     <metadata>
196         <title>Microsoft Internet Explorer 6 is installed</title>

```

```

197     <affected family="windows">
198         <platform>Microsoft Windows 2000</platform>
199         <platform>Microsoft Windows XP</platform>
200         <platform>Microsoft Windows Server 2003</platform>
201     </affected>
202     <reference source="CPE" ref_id="cpe:/a:microsoft:ie:6"/>
203     <description>The application Microsoft Internet Explorer 6 is installed.</description>
204     <oval_repository>
205         <dates>
206             <submitted date="2006-08-11T12:53:40">
207                 <contributor organization="ThreatGuard, Inc.">Robert L. Hollis</contributor>
208             </submitted>
209             <status_change date="2006-09-08T11:26:00.000-04:00">DRAFT</status_change>
210             <status_change date="2006-09-27T12:29:31.086-04:00">INTERIM</status_change>
211             <status_change date="2006-10-16T15:58:44.500-04:00">ACCEPTED</status_change>
212             <modified comment="Added an anchor to the regex used to check for Internet Explorer
213 6."
214         date="2007-01-11T20:38:00.950-05:00">
215             <contributor organization="The MITRE Corporation">Matthew Wojcik</contributor>
216         </modified>
217         <status_change date="2007-01-11T20:49:17.329-05:00">INTERIM</status_change>
218         <status_change date="2007-02-20T13:40:46.580-05:00">ACCEPTED</status_change>
219         <modified comment="Added CPE reference." date="2007-04-30T07:48:00.756-04:00">
220             <contributor organization="The MITRE Corporation">Jonathan Baker</contributor>
221         </modified>
222         <status_change date="2007-04-30T07:54:07.779-04:00">INTERIM</status_change>
223         <status_change date="2007-05-23T15:05:48.577-04:00">ACCEPTED</status_change>
224     </dates>
225     <status>ACCEPTED</status>
226 </oval_repository>
227 </metadata>
228 <criteria>
229     <criterion comment="Internet Explorer 6 (any patch level) is installed"
230     test_ref="oval:org.mitre.oval:tst:2333"/>
231 </criterion>
232 </definition>
233 <definition id="oval:org.mitre.oval:def:627" version="1" class="inventory">
234     <metadata>
235         <title>Microsoft Internet Explorer 7 is installed</title>
236         <affected family="windows">
237             <platform>Microsoft Windows XP</platform>
238             <platform>Microsoft Windows Server 2003</platform>
239             <platform>Microsoft Windows Vista</platform>
240         </affected>
241         <reference source="CPE" ref_id="cpe:/a:microsoft:ie:7"/>
242         <description>A version of Microsoft Internet Explorer 7 is installed.</description>
243         <oval_repository>
244             <dates>
245                 <submitted date="2007-01-09T06:00:00">
246                     <contributor organization="Secure Elements, Inc.">Sudhir Gandhe</contributor>
247                 </submitted>
248                 <status_change date="2007-01-11T15:30:00-04:00">DRAFT</status_change>
249                 <status_change date="2007-02-20T13:40:49.320-05:00">INTERIM</status_change>
250                 <modified comment="Added Microsoft Windows Vista to the list of affected platforms."
251                 date="2007-03-05T09:10:00.104-05:00">
252                     <contributor organization="The MITRE Corporation">Andrew Buttner</contributor>
253                 </modified>
254                 <status_change date="2007-03-21T16:17:23.092-04:00">ACCEPTED</status_change>
255             </dates>
256             <status>ACCEPTED</status>
257         </oval_repository>
258     </metadata>
259     <criteria>
260         <criterion comment="Internet Explorer 7 is installed"
261         test_ref="oval:org.mitre.oval:tst:178"
262         />
263     </criterion>
264 </definition>
265 </definitions>
266 <!-- ===== TESTS ===== -->
267 <!-- ===== TESTS ===== -->

```



```

268 <!-- ===== -->
269 <tests>
270   <registry_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
271     id="oval:gov.nist.fccc.patch:tst:57914" version="0" comment="Windows XP is installed"
272     check_existence="at_least_one_exists" check="at least one">
273     <object object_ref="oval:org.mitre.oval:obj:5590"/>
274     <state state_ref="oval:gov.nist.fccc.patch:ste:53828"/>
275   </registry_test>
276   <family_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
277     id="oval:org.mitre.oval:tst:99" version="1"
278     comment="the installed operating system is part of the Microsoft Windows family"
279     check_existence="at_least_one_exists" check="only one">
280     <object object_ref="oval:org.mitre.oval:obj:99"/>
281     <state state_ref="oval:org.mitre.oval:ste:99"/>
282   </family_test>
283   <registry_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
284     id="oval:org.mitre.oval:tst:178" version="1" comment="Internet Explorer 7 is installed"
285     check_existence="at_least_one_exists" check="at least one">
286     <object object_ref="oval:org.mitre.oval:obj:247"/>
287     <state state_ref="oval:org.mitre.oval:ste:115"/>
288   </registry_test>
289   <file_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
290     id="oval:org.mitre.oval:tst:427" version="1"
291     comment="the version of dhtmlcd.ocx is less than 6.1.0.9232"
292     check_existence="at_least_one_exists" check="all">
293     <object object_ref="oval:org.mitre.oval:obj:377"/>
294     <state state_ref="oval:org.mitre.oval:ste:394"/>
295   </file_test>
296   <registry_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
297     id="oval:org.mitre.oval:tst:1151" version="2"
298     comment="the patch kb891781 is installed (Hotfix key)"
299     check_existence="at_least_one_exists"
300     check="at least one">
301     <object object_ref="oval:org.mitre.oval:obj:823"/>
302     <state state_ref="oval:org.mitre.oval:ste:1031"/>
303   </registry_test>
304   <registry_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
305     id="oval:org.mitre.oval:tst:2333" version="2"
306     comment="Internet Explorer 6 (any patch level) is installed"
307     check_existence="at_least_one_exists" check="at least one">
308     <object object_ref="oval:org.mitre.oval:obj:247"/>
309     <state state_ref="oval:org.mitre.oval:ste:2185"/>
310   </registry_test>
311   <registry_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
312     id="oval:org.mitre.oval:tst:3019" version="1"
313     comment="Win2K/XP/2003 service pack 2 is installed" check_existence="at_least_one_exists"
314     check="at least one">
315     <object object_ref="oval:org.mitre.oval:obj:717"/>
316     <state state_ref="oval:org.mitre.oval:ste:2827"/>
317   </registry_test>
318   <file_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
319     id="oval:org.mitre.oval:tst:3422" version="3"
320     comment="the version of vxg.dll is less than 6.0.3790.4106"
321     check_existence="at_least_one_exists" check="at least one">
322     <object object_ref="oval:org.mitre.oval:obj:308"/>
323     <state state_ref="oval:org.mitre.oval:ste:3490"/>
324   </file_test>
325   <registry_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
326     id="oval:org.mitre.oval:tst:3653" version="2"
327     comment="a version of Windows for the x64 architecture is installed"
328     check_existence="at_least_one_exists" check="at least one">
329     <object object_ref="oval:org.mitre.oval:obj:1576"/>
330     <state state_ref="oval:org.mitre.oval:ste:3180"/>
331   </registry_test>
332   <registry_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
333     id="oval:org.mitre.oval:tst:3823" version="1"
334     comment="a version of Windows for the x86 architecture is installed"
335     check_existence="at_least_one_exists" check="at least one">
336     <object object_ref="oval:org.mitre.oval:obj:1576"/>
337     <state state_ref="oval:org.mitre.oval:ste:3649"/>
338   </registry_test>

```

```

339 <file_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
340   id="oval:org.mitre.oval:tst:3856" version="3"
341   comment="the version of vgx.dll is less than 6.0.2900.3164"
342   check_existence="at_least_one_exists" check="at least one">
343   <object object_ref="oval:org.mitre.oval:obj:308"/>
344   <state state_ref="oval:org.mitre.oval:ste:3185"/>
345 </file_test>
346 <file_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
347   id="oval:org.mitre.oval:tst:4182" version="3"
348   comment="the version of vgx.dll is less than 7.0.6000.20628"
349   check_existence="at_least_one_exists" check="at least one">
350   <object object_ref="oval:org.mitre.oval:obj:308"/>
351   <state state_ref="oval:org.mitre.oval:ste:3412"/>
352 </file_test>
353 <registry_test xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
354   id="oval:gov.nist.fdcc.patch:tst:115300" version="0"
355   comment="a version of Microsoft Windows XP is installed"
356   check_existence="at_least_one_exists"
357   check="at least one">
358   <object object_ref="oval:org.mitre.oval:obj:123"/>
359   <state state_ref="oval:gov.nist.fdcc.patch:ste:115263"/>
360 </registry_test>
361 </tests>
362 <!-- ===== -->
363 <!-- ===== OBJECTS ===== -->
364 <!-- ===== -->
365 <objects>
366   <family_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
367     id="oval:org.mitre.oval:obj:99" version="1"
368     comment="This is the default family object. Only one family object should exist."/>
369   <registry_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
370     id="oval:org.mitre.oval:obj:123" version="1"
371     comment="Registry key that hold the current windows os version">
372     <hive>HKEY_LOCAL_MACHINE</hive>
373     <key>SOFTWARE\Microsoft\Windows NT\CurrentVersion</key>
374     <name>CurrentVersion</name>
375   </registry_object>
376   <registry_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
377     id="oval:org.mitre.oval:obj:247" version="1"
378     comment="This registry key identifies the version of internet Explorer">
379     <hive>HKEY_LOCAL_MACHINE</hive>
380     <key>SOFTWARE\Microsoft\Internet Explorer</key>
381     <name>Version</name>
382   </registry_object>
383   <registry_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
384     id="oval:org.mitre.oval:obj:281" version="1"
385     comment="The registry key that identifies the location of the common files directory.">
386     <hive>HKEY_LOCAL_MACHINE</hive>
387     <key>SOFTWARE\Microsoft\Windows\CurrentVersion</key>
388     <name>CommonFilesDir</name>
389   </registry_object>
390   <file_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
391     id="oval:org.mitre.oval:obj:308" version="2">
392     <path var_ref="oval:org.mitre.oval:var:209" var_check="all"/>
393     <filename>vgx.dll</filename>
394   </file_object>
395   <file_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
396     id="oval:org.mitre.oval:obj:377" version="1">
397     <path var_ref="oval:org.mitre.oval:var:206" var_check="all"/>
398     <filename>dhtmlled.ocx</filename>
399   </file_object>
400   <registry_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
401     id="oval:org.mitre.oval:obj:717" version="1"
402     comment="This registry key holds the service pack installed on the host if one is
403   present.">
404     <hive>HKEY_LOCAL_MACHINE</hive>
405     <key>SOFTWARE\Microsoft\Windows NT\CurrentVersion</key>
406     <name>CSDVersion</name>
407   </registry_object>
408   <registry_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
409     id="oval:org.mitre.oval:obj:823" version="2">

```

```

410     <hive>HKEY_LOCAL_MACHINE</hive>
411     <key operation="equals">SOFTWARE\Microsoft\Windows NT\CurrentVersion\Hotfix\KB891781</key>
412     <name operation="equals">IsInstalled</name>
413   </registry_object>
414   <registry_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
415     id="oval:org.mitre.oval:obj:1576" version="1"
416     comment="This registry key identifies the architecture on the system">
417     <hive>HKEY_LOCAL_MACHINE</hive>
418     <key>SYSTEM\CurrentControlSet\Control\Session Manager\Environment</key>
419     <name>PROCESSOR_ARCHITECTURE</name>
420   </registry_object>
421   <registry_object xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
422     id="oval:org.mitre.oval:obj:5590" version="1" comment="This registry key ProductName">
423     <hive>HKEY_LOCAL_MACHINE</hive>
424     <key>SOFTWARE\Microsoft\Windows NT\CurrentVersion</key>
425     <name>ProductName</name>
426   </registry_object>
427 </objects>
428 <!-- ===== -->
429 <!-- ===== STATES ===== -->
430 <!-- ===== -->
431 <states>
432   <registry_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
433     id="oval:gov.nist.fccc.patch:ste:53828" version="0" comment="The registry key matches with
434 XP">
435     <value operation="pattern match">.*[XPxp].*</value>
436   </registry_state>
437   <family_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
438     id="oval:org.mitre.oval:ste:99" version="1" comment="Microsoft Windows family">
439     <family>windows</family>
440   </family_state>
441   <registry_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
442     id="oval:org.mitre.oval:ste:115" version="1"
443     comment="The registry key has a value that matches 7.*">
444     <value operation="pattern match">^7\..*${</value>
445   </registry_state>
446   <registry_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
447     id="oval:org.mitre.oval:ste:1031" version="1">
448     <value operation="equals" datatype="int">1</value>
449   </registry_state>
450   <registry_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
451     id="oval:org.mitre.oval:ste:2185" version="2"
452     comment="The registry key has a value that matches 6.*">
453     <value operation="pattern match">^6\..*${</value>
454   </registry_state>
455   <registry_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
456     id="oval:org.mitre.oval:ste:2827" version="1"
457     comment="The registry key has a value of Service Pack 2">
458     <value>Service Pack 2</value>
459   </registry_state>
460   <registry_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
461     id="oval:org.mitre.oval:ste:3180" version="2">
462     <value>amd64</value>
463   </registry_state>
464   <file_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
465     id="oval:org.mitre.oval:ste:3185" version="2">
466     <version operation="less than" datatype="version">6.0.2900.3164</version>
467   </file_state>
468   <file_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
469     id="oval:org.mitre.oval:ste:3412" version="2">
470     <version operation="less than" datatype="version">7.0.6000.20628</version>
471   </file_state>
472   <file_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
473     id="oval:org.mitre.oval:ste:3490" version="2">
474     <version operation="less than" datatype="version">6.0.3790.4106</version>
475   </file_state>
476   <registry_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
477     id="oval:org.mitre.oval:ste:3649" version="1" comment="x86 architecture">
478     <value>x86</value>
479   </registry_state>
480   <file_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"

```

```

481     id="oval:org.mitre.oval:ste:394" version="1">
482       <version datatype="version" operation="less than">6.1.0.9232</version>
483     </file_state>
484     <registry_state xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
485       id="oval:gov.nist.fdcc.patch:ste:115263" version="0"
486       comment="The registry key has a value of 5.1 or 5.2">
487       <value operation="pattern match">5\.(1|2)</value>
488     </registry_state>
489   </states>
490   <!-- ===== -->
491   <!-- ===== VARIABLES ===== -->
492   <!-- ===== -->
493   <variables>
494     <local_variable id="oval:org.mitre.oval:var:206" version="1"
495       comment="Windows common files\microsoft shared\triedit directory" datatype="string">
496       <concat>
497         <object_component item_field="value" object_ref="oval:org.mitre.oval:obj:281"/>
498         <literal_component>\microsoft shared\triedit</literal_component>
499       </concat>
500     </local_variable>
501     <local_variable id="oval:org.mitre.oval:var:209" version="2"
502       comment="Base path to vgx.dll, part of Vector Markup Language (VML) implementation."
503       datatype="string">
504       <concat>
505         <object_component item_field="value" object_ref="oval:org.mitre.oval:obj:281"/>
506         <literal_component>\Microsoft Shared\VGX</literal_component>
507       </concat>
508     </local_variable>
509   </variables>
510   <!-- ===== -->
511   <!-- ===== -->
512   <!-- ===== -->
513 </oval_definitions>

```

## C.4 OCIL Questionnaire

The following OCIL XML instance represents a questionnaire used to evaluate the XCCDF checklist from the previous section.

**Figure C-4. example-winxp-ocil.xml**

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <ocil xsi:schemaLocation="http://www.mitre.org/ocil/2 ocil.xsd"
3  xmlns="http://www.mitre.org/ocil/2"
4  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
5
6    <generator>
7      <schema_version>2.0</schema_version>
8      <timestamp>2009-09-01T11:34:01</timestamp>
9      <author>
10         <name>John Doe</name>
11      </author>
12    </generator>
13
14    <document>
15      <title>Example Win XP OCIL Content</title>
16      <description>The following contains checks for FDCC rules that cannot be fully automated with
17        existing products.
18      </description>
19    </document>
20
21    <questionnaires>
22      <questionnaire id="ocil:mitre.org:questionnaire:1">
23        <title>Network access: Allow anonymous SID/Name translation</title>
24        <references>
25          <reference href="cce.mitre.org/version/4">CCE-953</reference>
26          <reference href="http://cce.mitre.org">CCE-2973-6</reference>
27        </references>
28        <actions>
29          <test_action_ref>ocil:mitre.org:testaction:51</test_action_ref>
30        </actions>
31      </questionnaire>
32    </questionnaires>
33
34    <test_actions>
35      <boolean_question_test_action question_ref="ocil:mitre.org:question:51"
36        id="ocil:mitre.org:testaction:51">
37        <when_true>
38          <result>PASS</result>
39        </when_true>
40        <when_false>
41          <result>FAIL</result>
42        </when_false>
43      </boolean_question_test_action>
44    </test_actions>
45
46    <questions>
47      <boolean_question id="ocil:mitre.org:question:51">
48        <question_text>Is anonymous SID / Name translation allowed?</question_text>
49      </boolean_question>
50    </questions>
51  </ocil>
52

```

## C.5 CPE Dictionary

This minimal CPE dictionary XML instance contains CPE Names referenced in the XCCDF document presented in section C.1.

**Figure C-5. example-winxp-cpe-dictionary.xml**

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <cpe-list xmlns="http://cpe.mitre.org/dictionary/2.0"
3     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xsi:schemaLocation="http://cpe.mitre.org/dictionary/2.0 http://cpe.mitre.org/files/cpe-
5 dictionary_2.1.xsd">
6     <cpe-item name="cpe:/o:microsoft:windows_xp">
7         <title>Microsoft Windows XP</title>
8         <check system="http://oval.mitre.org/XMLSchema/oval-definitions-5" href="example-
9 winxp-cpe-oval.xml">oval:gov.nist.fdcc.xp:def:2</check>
10     </cpe-item>
11 </cpe-list>

```

## C.6 CPE Inventory

The following OVAL XML instance contains OVAL Definitions that SHOULD be used to evaluate the CPE Name defined in the previous section.

**Figure C-6. example-winxp-cpe-oval.xml**

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <oval_definitions xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5"
3     xmlns:oval="http://oval.mitre.org/XMLSchema/oval-common-5"
4     xmlns:oval-def="http://oval.mitre.org/XMLSchema/oval-definitions-5"
5     xmlns:win-def="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows"
6     xmlns:ind-def="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
7     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
8     xsi:schemaLocation="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows
9 http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/windows-
10 definitions-schema.xsd
11 http://oval.mitre.org/XMLSchema/oval-definitions-5#independent
12 http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/independent-
13 definitions-schema.xsd
14 http://oval.mitre.org/XMLSchema/oval-definitions-5
15 http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/oval-
16 definitions-schema.xsd
17 http://oval.mitre.org/XMLSchema/oval-common-5
18 http://oval.mitre.org/language/download/schema/version5.3/ovaldefinition/complete/oval-common-
19 schema.xsd">
20     <generator>
21         <oval:product_name>National Institute of Standards and Technology</oval:product_name>
22         <oval:schema_version>5.6</oval:schema_version>
23         <oval:timestamp>2008-09-02T12:59:10.000-04:00</oval:timestamp>
24     </generator>
25     <!-- ===== DEFINITIONS ===== -->
26     <!-- ===== DEFINITIONS ===== -->
27     <!-- ===== DEFINITIONS ===== -->
28     <definitions>
29         <definition id="oval:gov.nist.fdcc.xp:def:2" version="1" class="inventory">
30             <metadata>
31                 <title>Microsoft Windows XP is installed</title>
32                 <affected family="windows">
33                     <platform>Microsoft Windows XP</platform>
34                 </affected>
35                 <description>Microsoft Windows XP is installed</description>
36             </metadata>
37             <criteria>
38                 <criterion comment="the installed operating system is part of the
39 Microsoft Windows family" test_ref="oval:gov.nist.fdcc.xp:tst:6"/>
40                 <criterion comment="Microsoft Windows XP is installed"
41 test_ref="oval:gov.nist.fdcc.xp:tst:7"/>
42             </criteria>
43         </definition>
44     </definitions>
45     <!-- ===== TESTS ===== -->
46     <!-- ===== TESTS ===== -->
47     <!-- ===== TESTS ===== -->
48     <tests>
49         <family_test id="oval:gov.nist.fdcc.xp:tst:6" version="1" comment="the installed

```

```

50 operating system is part of the Microsoft Windows family" check_existence="at_least_one_exists"
51 check="only one" xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent">
52     <object object_ref="oval:gov.nist.fdcc.xp:obj:3"/>
53     <state state_ref="oval:gov.nist.fdcc.xp:ste:14"/>
54 </family_test>
55 <registry_test id="oval:gov.nist.fdcc.xp:tst:7" version="1" comment="Microsoft
56 Windows XP is installed" check_existence="at_least_one_exists" check="at least one"
57 xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows">
58     <object object_ref="oval:gov.nist.fdcc.xp:obj:4"/>
59     <state state_ref="oval:gov.nist.fdcc.xp:ste:15"/>
60 </registry_test>
61 </tests>
62 <!-- ===== -->
63 <!-- ===== OBJECTS ===== -->
64 <!-- ===== -->
65 <objects>
66     <family_object id="oval:gov.nist.fdcc.xp:obj:3" version="1"
67 xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"/>
68     <registry_object id="oval:gov.nist.fdcc.xp:obj:4" version="1"
69 xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows">
70         <hive>HKEY_LOCAL_MACHINE</hive>
71         <key>SOFTWARE\Microsoft\Windows NT\CurrentVersion</key>
72         <name>CurrentVersion</name>
73     </registry_object>
74 </objects>
75 <!-- ===== -->
76 <!-- ===== STATES ===== -->
77 <!-- ===== -->
78 <states>
79     <family_state id="oval:gov.nist.fdcc.xp:ste:14" version="1"
80 xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#independent">
81         <family>windows</family>
82     </family_state>
83     <registry_state id="oval:gov.nist.fdcc.xp:ste:15" version="1"
84 xmlns="http://oval.mitre.org/XMLSchema/oval-definitions-5#windows">
85         <value>5.1</value>
86     </registry_state>
87 </states>
88 <!-- ===== -->
89 <!-- ===== -->
90 <!-- ===== -->
91 </oval_definitions>

```